Clasificación por Dialecto de Documentos Escritos en Quechua

por

Rosemary Jiménez Medina

Proyecto sometido en cumplimiento parcial de los requisitos para optar el grado de

Master en Ingeniería Computacional y Sistemas Inteligentes

Universidad del País Vasco

Euskal Herriko Unibertsitatea

Universidad Nacional de San Antonio Abad del Cusco

Directores: Olatz Arregui

Iñaki Alegría

Noviembre 2013

ABSTRACT

In this work we construct the first automatic text classifier written in Quechua. This will be a binary classifier to determine the dialect of the text (Cusco, Cusco not). They represent the characteristics of the corpus texts using as attributes: words, lemmas, bigrams, trigrams and combinations of all these. The techniques of dimensionality reduction are the *stop word list* and *stemming*. Will use different types of classification algorithms: decisions trees, rules, Support Vector Machine and Naive Bayes.

This work has been inspired by the text classification work performed for Basque by the IXA group at the University of the Basque Country, language like Quechua is binding.

The results obtained when evaluating the different classifiers are encouraging and show the advantages of pre-processes applied at different stages.

RESUMEN

En este trabajo se construirá un primer clasificador automático de textos escritos en quechua. Este será un clasificador de tipo binario para determinar el dialecto del texto (cusqueño, no cusqueño). Se representarán las características de los textos del corpus empleando como atributos: palabras, lemas, bigramas, trigramas y las combinaciones de todos éstos. Se utilizarán como técnicas de reducción de la dimensionalidad el *stop word list* y la *lematización*. Emplearemos diferentes tipos de algoritmos de clasificación: árboles de decisión, reglas, *Support Vector Machine* y *Naive Bayes*.

Este trabajo ha sido inspirado por los trabajos de clasificación de textos realizados para el euskera por el grupo IXA de la Universidad del Pais Vasco, lengua que al igual que el quechua es aglutinante.

Los resultados obtenidos al evaluar los distintos clasificadores son alentadores y muestran las ventajas de los pre-procesos aplicados en sus distintas fases.

TABLA DE CONTENIDO

LISTA DE TABLAS

LISTA DE FIGURAS

1 INTRODUCCIÓN

El problema de la clasificación automática de textos tiene una larga historia, que se remonta al menos a 1960. En los años 80 el problema fue abordado con el enfoque de la ingeniería del conocimiento, que consistía en la construcción manual de un sistema experto capaz de tomar decisiones de categorización. Tal sistema experto consistía en un conjunto de reglas definidas manualmente. El inconveniente de este enfoque era la existencia de un cuello de botella en la adquisición del conocimiento, las reglas debían ser definidas manualmente por un ingeniero del conocimiento con la ayuda de un experto del dominio. En los años 90, gracias a la disponibilidad de documentos en línea, aumentó el interés sobre el problema de clasificación automática de textos, y es así como aparece un nuevo paradigma para abordarlo basado en aprendizaje automático (*machine learning, ML*).

Cuando se habla de clasificación automática se distingue entre dos escenarios diferentes, que requieren soluciones distintas. Estos escenarios reciben diversos nombres, pero básicamente consisten en lo siguiente: de un lado, una situación en la que se parte de una serie de clases o categorías conceptuales prediseñadas a priori, y en la que labor del clasificador (manual o automático) es asignar cada documento a la clase o categoría que le corresponda. Es lo que se conoce como clasificación supervisada o categorización, no sólo porque requiere la elaboración manual o intelectual del cuadro o esquema de categorías, sino también, porque requiere un proceso de aprendizaje o entrenamiento por parte del clasificador, que debe ser supervisado manualmente en mayor o menor medida.

En el segundo escenario posible, no hay categorías previas ni esquemas o cuadros de clasificación establecidos a priori. Los documentos se agrupan en función de ellos mismos, de su contenido; de alguna

manera, podemos decir que se autoorganizan. Es lo que se conoce como clasificación automática no supervisada (*Clustering*); no supervisada porque se efectúa de forma totalmente automática, sin supervisión o asistencia manual [18].

El objetivo principal de este proyecto es obtener un clasificador automático por dialecto de documentos escritos en quechua. Para ello se utilizarán las técnicas de *Machine Learning*. Dicho clasificador determinará si un documento está escrito en quechua cusqueño o no. Partiremos de un conjunto de documentos previamente clasificados manualmente, para después intentar buscar el algoritmo y los atributos que mejor se adecuen a las características del lenguaje.

Para llevar a cabo todo el proceso se han identificado las tareas específicas que a continuación se presentan:

- Recopilar un corpus de texto escrito en quechua cusqueño junto con documentos escritos en quechua no cusqueño, en los diferentes formatos de documentos como .pdf, .doc, contenido web e incluso en formato no electrónico.
- Preprocesar los documentos de texto del corpus, para prepararlos para el siguiente paso de la clasificación. Para ello habrá que abordar diferentes subtareas como eliminar caracteres y palabras erróneas, aplicar filtros de frecuencia de aparición y representar los documentos con las características que se consideren representativas del lenguaje.
- Seleccionar los atributos que mejor se adecuen al dialecto y representen las características, del mismo.
- Seleccionar los algoritmos que mejores resultados proporcionen en la clasificación por dialecto de los documentos escritos en quechua.

Este trabajo está organizado en 8 capítulos. El segundo capítulo presenta los antecedentes y el estado actual de la clasificación automática de documentos.

En el tercer capítulo se detallan las características específicas de la lengua quechua; su expansión por Sudamérica, sus distintas variedades y tipología.

El cuarto capítulo explica en qué consiste la tarea de la clasificación automática de textos; cómo se define formalmente esta tarea y las fases que la componen; la fase de indexación y reducción de la dimensionalidad, la fase de la construcción del clasificador y la fase de la evaluación del clasificador. Además se citan algunas técnicas asociadas a cada fase.

Seguidamente en el capítulo quinto se detalla la tarea de la recopilación del corpus de documentos escritos en quechua cusqueño y no cusqueño; las características de este corpus y las tareas de preprocesado de los documentos de texto.

El capítulo sexto puntualiza la experimentación llevada a cabo; por qué se utilizó Weka como principal herramienta, que algoritmos se utilizaron para la construcción de los clasificadores, cuáles fueron los atributos que se escogieron y la descripción de cómo y en qué orden se realizaron de los experimentos.

Posteriormente en el capítulo séptimo se mostrarán los valores de los resultados logrados y se describirá una evaluación para cada uno de ellos.

Finalmente en el capítulo octavo, se explican las conclusiones a las cuales se arribó tras finalizar el presente trabajo, y las tareas futuras a realizar en este ámbito.

2 ANTECEDENTES Y ESTADO ACTUAL

El problema de la clasificación automática de documentos ha sido abordado con diversos métodos. Tal y como se mencionó en el capítulo 1, la historia de la clasificación automática de textos se remonta al menos a 1960, pero es desde los años 80 que el problema fue abordado con la construcción sistemas expertos, los cuales consistían en un conjunto de reglas definidas manualmente por los denominados ingenieros del conocimiento.

En los últimos años la mayor parte de los métodos empleados para la clasificación automática de textos, se basan en las técnicas de aprendizaje automático (*machine learning*).

Los sistemas de aprendizaje automático necesitan un conjunto de datos de entrenamiento o aprendizaje de donde se obtiene una generalización inductiva que se utiliza para el aprendizaje del sistema, de esta manera se genera el clasificador.

Para poder llevar a cabo la clasificación automática es preciso contar con una serie de elementos previos. En nuestro caso, para empezar, lo más importante es contar con una forma consistente de representar cada documento, es decir su contenido. La mayoría de los estudios realizados, han empleado una clasificación basada en la representación del documento a clasificar por las palabras que lo componen, denominada *bag of words,* como se puede apreciar en [9] y [16]. También existen otros trabajos que analizan el impacto que puede ocasionar la utilización de otro tipo de características para representar el documento que se intenta categorizar, como pueden ser los lemas, los sintagmas y los N-gramas.

En [20], se presenta un método muy simple basado en principios estadísticos, con el uso de secuencias de caracteres denominados N-

gramas, para la categorización por lenguaje, sólo para textos cortos. En [15] se muestra una mejora del método de representación de textos a través de N-gramas, la diferencia es que esta vez es especialmente utilizado para trabajar con textos largos. En [13] se emplean N-gramas para identificar el idioma de los textos.

Otra técnica básica y muy importante para la reducción de la dimensionalidad, es la *lematización* de las palabras que componen el corpus. La lematización consiste en la eliminación de sufijos y afijos de una palabra, de tal modo que aparezca sólo su raíz léxica denominada lema de la palabra. En [6] y [17], se señala cómo puede influir el uso de algoritmos de *stemming* en la clasificación de documentos. En [14] se muestra el impacto de la lematización en la clasificación de documentos escritos en euskera, lengua que es aglutinante al igual que el quechua.

Por último, otra técnica de reducción de la dimensionalidad del corpus, que resulta útil para mejorar el rendimiento de los algoritmos de clasificación, es la aplicación de *stop word list* para eliminar las palabras funcionales, aquellas que no transmiten información como es el caso de pronombres, preposiciones, conjunciones, etc.

3 ESPECIFICIDAD DEL QUECHUA

3.1 La familia lingüística Quechua

El quechua es una familia lingüística y no una lengua. Se afirma ésto con el hecho de que son muchas las maneras en que se habla y que estas maneras pueden ser emparentadas en dos grandes grupos, cuyas diferencias hacen que casi no sean inteligibles entre sí. Estos dos grandes grupos son el quechua central y el quechua sureño/norteño (curiosamente, en muchos sentidos, las hablas norteñas son más parecidas a las del sur que a las del centro).

La familia lingüística quechua se extiende desde el sur de Colombia hasta el norte de Chile y Argentina. Abarca los países de Colombia, Ecuador, Perú, Bolivia, Chile y Argentina. Además del quechua de los Andes, también se hablan variedades quechuas en algunas áreas de la ceja de selva, y de la selva baja del Perú y el Ecuador. La mayor extensión geográfica y la mayor diversidad de idiomas quechuas se encuentran en el Perú.

A menudo se sostiene que el quechua del Cusco es la lengua materna de todos los quechua hablantes. Muchos creen que el quechua es un solo idioma con muchos dialectos regionales. El quechua es una familia lingüística con una variedad de idiomas, muchos de los cuales tienen dialectos dentro de la región geográfica donde se habla. Según Alfredo Torero (1964), las formas más antiguas del quechua, las primigenias, se hablan en Ancash, Cerro de Pasco, Junín, Huánuco y la sierra de Lima. “Las variedades de la familia lingüística quechua se clasifican en dos grandes grupos de idiomas: I) las habladas en Ancash, Huánuco, Pasco, Junín y algunas provincias del departamento de Lima, y II)

las habladas en Apurímac, Huancavelica, Ayacucho, Cusco, Puno, Lambayeque, Cajamarca, San Martín, Amazonas y algunas zonas de Loreto" [10].

Las diferencias que existen entre los idiomas quechuas son comparables con las diferencias que existen entre el castellano, el portugués, el francés y el italiano. Muchos de los quechua hablantes no se refieren a su idioma como "*quechua*". Los nombres varían según la zona. Por ejemplo, en Colombia se llama *inga*, en San Martín *llakwash*, en Cajamarca *lingwa*.

En el sur del Perú el nombre que se le da es *runa simi* o *runa shimi*, (*runa* "hombre", *simi* "lengua, boca") que significa 'idioma del hombre' o 'boca del hombre'. Sin embargo, en la actualidad los quechua hablantes del departamento de Huánuco y de algunos otros lugares se refieren a su idioma como "*quechua*" o "*gechwa*".

Parece que la palabra "*quechua*" surgió de un error cometido por los españoles que tomaron la palabra *qheswa* 'valle' de la denominación del dialecto *qheswa simi* que significa 'idioma del valle', para referirse al idioma [10].

En los Andes del Perú los quechua hablantes han incorporado a su lengua varias palabras castellanas. Es un fenómeno que se encuentra en todos los idiomas vivos, que poco a poco incrementan su propio vocabulario con préstamos de otros idiomas. También en la selva peruana, algunos idiomas han incorporado muchas palabras quechuas. Este fenómeno está cambiando, ya que se están incorporando más préstamos del castellano en el proceso de crecimiento y desarrollo.

Figura 3.1: Extensión de la familia lingüística quechua.

Como se ha mencionado anteriormente, la familia lingüística quechua se clasifica en dos grandes grupos (I y II) dependiendo de la zona donde se hable, el grupo I es el quechua hablado en la región central del Perú y el quechua II es el hablado en el norte y en el sur del Perú. A su vez el grupo II se subdivide en 3 subgrupos (A, B y C) como se muestra con más detalle a continuación.

Variedades del quechua habladas en el Perú:

Grupo I
1. Ambo-Pasco
2. Chiquián-Cajatambo
3. Conchucos, Norte de
4. Conchucos, Sur de
5. Corongo
6. Huallaga
7. Huamalíes
8. Huaylas
9. Junín, Norte de
10. Margos-Lauricocha-Yarowilca
11. Pacaraos
12. Pachitea
13. Sihuas
14. Wanca

Grupo II A
15. Cajamarca
16. Chachapoyas
17. Lambayeque

Grupo II B
18. Napo
19. Pastaza
20. San Martín
21. Santarrosino
22. Tigre

Grupo II C
23. Apurímac
24. Ayacucho
25. Cusco-Collao
26. La Unión

En el mapa del Perú que se muestra en la figura 3.2, podemos ubicar las variedades del quechua antes mencionadas.

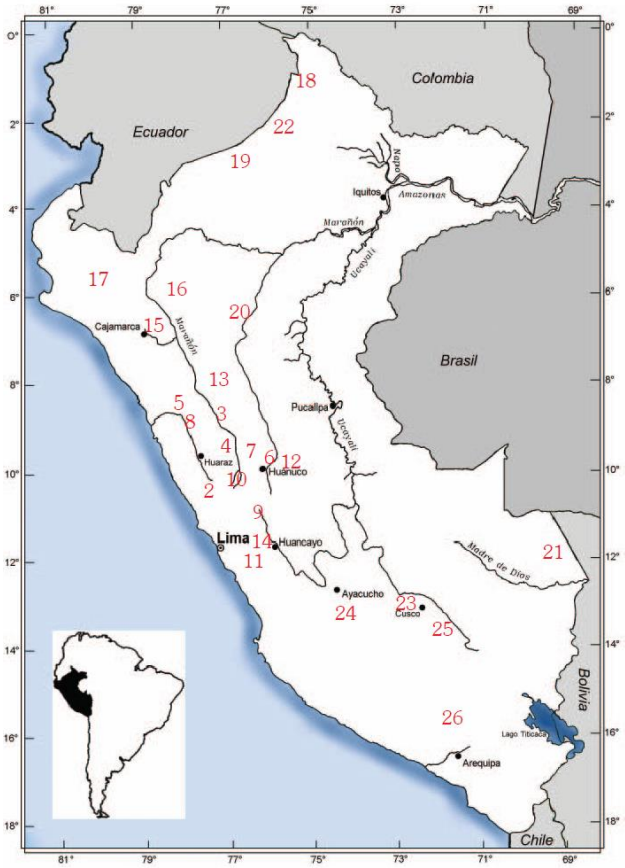

Figura 3.2: Mapa de variedades de quechua hablantes en el Perú.

Como se ha visto el quechua muestra un variado abanico de distintas variedades de lenguas quechuas. Después de la consulta de una cierta cantidad de trabajos sobre esta lengua andina nos percatamos de que no es posible hablar de un único runa simi.

A continuación se presentan algunas de las características específicas del lenguaje quechua, debido a que el conocimiento de estas características puede ser de gran importancia para la clasificación automática de documentos quechua.

3.2 Tipología particular del Quechua

3.2.1 Aglutinante

Según la terminología lingüística, el quechua es una lengua sufijal y aglutinante, lo cual quiere decir que las palabras se forman mediante la adición a la raíz de múltiples pequeñas partículas denominadas terminaciones o sufijos, que no cambian mayormente su forma al combinarse dentro de una palabra. Por ejemplo:

wasi 'casa'

wasi-cha 'casita'

wasi-cha-yki 'tu casita'

wasi-cha-yki-chik 'su casita (de ustedes)'

wasi-cha-yki-chik-kuna 'sus casitas (de ustedes)'

wasi-cha-yki-chik-kuna-paq 'para sus casitas (de ustedes)'

wasi-cha-yki-chik-kuna-paq-chá 'tal vez para sus casitas (de ustedes)'

El ejemplo anterior corresponde a una raíz sustantiva, pero se puede hacer lo propio con las raíces verbales. Por ejemplo:

yacha 'saber'

yacha-chi 'hacer saber, enseñar'

yacha-chi-naya 'querer enseñar'

yacha-chi-naya-chka 'estar queriendo enseñar'

yacha-chi-ku-chka-n 'él/ella está queriendo enseñar'

yacha-chi-ku-chka-n-ku 'ellos/ellas están queriendo enseñar'

En la lengua quechua, la morfología derivativa se impone sobre la flexiva [12].

3.2.2 Orden de palabras

La lengua quechua privilegia en sus oraciones el orden sujeto – objeto – verbo (SOV).

Ejemplos:

Alqo aychata mijun (El perro carne come)
S O V

Waka sarata mijuramuska (La vaca maíz había comido)
S O V

Julia llamakunata michin (Julia las llamas pastea)
S O V

3.2.3 Número de vocales utilizadas

El único estándar que existe para el quechua cusqueño es el estándar denominado "Quechua Sureño Unificado" estándar literario creado por Rodolfo Cerrón Palomino para los dialectos del sur de la zona quechuahablante. Estándar con el cual la

Academia Mayor de la Lengua Quechua, no está de acuerdo, esto debido a que dicho estándar plantea sólo el uso de 3 vocales (a, i, u) en lugar de 5 (a, e, i, o, u).

El problema del trivocalismo quechua pasó inadvertido para una mayoría de los estudiosos de esa lengua durante más o menos cuatrocientos cincuenta años. En el decurso de ese tiempo se aceptó la escritura y vocalización del quechua utilizando las cinco vocales del castellano. El enredo del penta y trivocalismo emergió al mundillo lingüístico con ocasión de la promulgación de la Ley Nro. 21156 del 27 de mayo de 1975, que oficializó el quechua en el Perú con rango equivalente al de la lengua castellana [1].

Los primeros planteamientos sobre el trivocalismo arrancan en el libro "Gramática de la Lengua General del Perú" llamada comúnmente quichua, de Fray Miguel Angle Mossi publicado en la ciudad de Sucre (Bolivia) en 1806, cuando plantea que el vocalismo quechua está formado por un sistema triangular de dos vocales altas "I U" y una baja "A", desechando las vocales "e", "o". Punto de vista que se reitera en 1975. Mas el primer texto editado en Valladolid 1560, por Fray Domingo de Santo Tomás, con el título de "Gramática o Arte de la Lengua General de los Indios de los Reinos del Perú", adaptó la signo grafía castellana a la fonética Quechua, con los siguientes valores vocálicos: a, e, i, o, u.

3.3 Diferencias dialectales

La diferencia clave, pero no la única, que ha permitido separar al quechua central (I) del quechua sureño/norteño (II), tiene que ver con la manera como ambas variedades conjugan la primera persona singular de sus verbos.

La diferencia fundamental que existe entre el quechua central (I) y el sureño/norteño (II) es que el segundo emplea -ni para conjugar la primera persona singular en presente y el primero no. Sin embargo, como se ha mencionado inicialmente, esto no significa que no haya más diferencias entre los distintos quechuas; por el contrario las diferencias entre unas formas de hablar y otras son muy amplias y alcanzan, inclusive, a los dialectos de una misma rama. Por ejemplo, dentro del quechua central (I) se distinguen 12 variedades y pueden haber diferencias entre ellas como se ve en la tabla 3.1 (el signo /:/ significa que la vocal anterior se alarga):

QUECHUA CENTRAL		QUECHUA SUREÑO/NORTEÑO	CASTELLANO
Junín/Ancash	Pacaraos	Todos los dialectos sin excepción	
puri-:	*puri-y*	*puri-ni*	"yo camino"
miku-:	*miku-y*	*miku-ni*	"yo como"
upya-:	*upya-y*	*upya-ni*	"yo bebo"

Tabla 3.1: Conjugación en quechua central (I) y quechua sureño/norteño (II) de la primera persona singular.

Se ve como a pesar de que las variedades del quechua de Pacaraos(11) y de Junín/Ancash (2, 3, 4, 8), ambos pertenecientes al quechua central (I), existen inclusive diferencias entre estos dos.

A continuación en la tabla 3.2, se podrá apreciar las distintas formas que hay en algunas regiones dialectales del quechua, para algunas palabras de más uso [10]:

Quechua		**Lugar**	**Palabra**							
			padre	madre	tierra	agua	sol	Luna	hombre	mujer
Grupo I		Ambo-Pasco	*tayta (formal), papä (familiar)*	*mama*	*pacha*	*yacu*	*inti*	*quilla*	*runa*	*warmi*
		Chiquián-Cajatambo	*tayta*	*mama*	*pasa*	*yacu*	*inti*	*quila*	*nuna*	*warmi*
		Conchucos	*taytay*	*mamay*	*patsa*	*yacu*	*rupay*	*killa*	*ollgo o runa*	*warmi*
		Sur de Conchucos	*tayta*	*mama*	*patsa*	*yacu*	*rupay/ inti*	*quilla*	*runa*	*warmi*
		Corongo	*tëta*	*mama*	*patsa*	*yacu*	*rupë*	*quilla*	*runa*	*warmi*
		Huallaga	*tayta*	*mama*	*allpa*	*yacu*	*inti*	*quilla*	*Runa*	*warmi*
		Huamalíes	*tayta*	*mamá*	*alpa*	*yacu*	*inti*	*quila*	*Runa*	*warmi*
		Huaylas	*yaya*	*mama*	*patsa*	*yacu*	*inti*	*killa*	*nuna*	*warmi*
		Norte de Junín	*tayta*	*mama*	*alpa*	*yacu*	*inti*	*quilla*	*Olgu*	*warmi*
		Margos-Yarowilca-Lauricocha	*papä*	*mama*	*alpa*	*yacu*	*inti*	*quilla*	*runa*	*warmi*
		Panao-Pachitea	*tayta*	*mama*	*pacha*	*yacu*	*inti*	*quilla*	*runa*	*warmi*
		Wanca	*tayta*	*mama*	*allpa*	*yacu*	*inti*	*quilla*	*wayapa*	*walmi*
Grupo II	**A**	Cajamarca	*tayta*	*mama*	*pacha*	*yaku*	*rupay*	*killa*	*runa*	*warmi*
		Lambayeque	*taytay*	*mamay*	*pacha*	*yaku*	*rupay*	*killa*	*runa*	*warmi*
	B	Napo	*yaya*	*mama*	*allpa*	*yaku*	*inti*	*killa*	*kari*	*warmi*
		Pastaza	*yaya*	*mama*	*allpa*	*yaku*	*inti*	*killa*	*kari*	*warmi*
		San Martín	*tata*	*mama*	*allpa*	*yaku*	*inti*	*killa*	*ullku runa*	*warmi*
	C	Apurímac	*papá*	*mama*	*allpa*	*unu*	*inti*	*killa*	*qari*	*warmi*
		Arequipa	*tayta*	*mama*	*hallp'a*	*yaku*	*inti*	*killa*	*qari*	*warmi*
		Ayacucho	*tayta*	*mama*	*allpa*	*yaku*	*inti*	*killa*	*qari*	*warmi*
		Cusco-Collao	*tayta*	*mama*	*hallp'a*	*unu*	*inti*	*killa*	*qhari*	*warmi*

Tabla 3.2: Diferencia de palabras quechua en las distintas regiones dialectales.

Una diferencia marcada entre el grupo I y II, es el uso de la palabra “agua”: *yacu* para el caso del quechua I y *yaku* (mayoritariamente) y *unu* para el caso del quechua II. Otra diferencia notoria entre ambos grupos se presenta con la palabra “luna”: *quilla* (mayoritariamente) en el caso del grupo I y *killa* en el caso del quechua II.

De los ejemplos antes mencionados, se puede notar el uso de la consonante “k” en el quechua II en lugar de “c” y “qu” empleado en el quechua I.

4 CLASIFICACIÓN DE DOCUMENTOS

La clasificación de textos es un término ampliamente utilizado para la clasificación de documentos. La Figura 4.1 ilustra la aplicación de clasificación de documentos. Los documentos están organizados en carpetas, una carpeta para cada tema. Un nuevo documento se presenta, y el objetivo es poner este documento en la carpeta adecuada [19].

Figura 4.1: Categorización de textos.

La clasificación de documentos se puede ver como la tarea de determinar una asignación de un valor {0,1} para cada entrada de la matriz de decisión [7].

	$\boldsymbol{d_1}$	⋯	⋯	$\boldsymbol{d_j}$	⋯	⋯	$\boldsymbol{d_n}$
$\boldsymbol{c_1}$	a_{11}	⋯	⋯	a_{1j}	⋯	⋯	a_{1n}
⋯	⋯	⋯	⋯	⋯	⋯	⋯	⋯
$\boldsymbol{c_i}$	a_{i1}	⋯	⋯	a_{ij}	⋯	⋯	a_{in}
⋯	⋯	⋯	⋯	⋯	⋯	⋯	⋯
$\boldsymbol{c_m}$	a_{m1}	⋯	⋯	a_{mj}	⋯	⋯	a_{mn}

Donde $C = \{c_1, \dots, c_m\}$ es un conjunto de categorías predefinidas, y $D = \{d_1, \dots, d_n\}$ es un conjunto de documentos que se han de categorizar. Un valor de 1 para a_{ij} es interpretado como una decisión para presentar d_j bajo c_i, mientras que un valor de 0 se interpreta como una decisión para no presentar d_j bajo c_i.

Es fundamental para la comprensión de esta tarea las siguientes dos observaciones:

- las categorías son sólo etiquetas simbólicas. No hay conocimiento adicional disponible de su "sentido" para ayudar en el proceso de construcción del clasificador.
- la atribución de los documentos a categorías debe ser efectuado sobre la base del contenido de estos documentos [7].

El enfoque de aprendizaje automático es un proceso inductivo. En general crea automáticamente un clasificador para una categoría c_i por medio de la "observación" de las características de un conjunto de documentos que han sido previamente clasificadas manualmente en c_i por un experto de dominio, a partir de estas características, el proceso inductivo recolecta la características que un documento nuevo debe tener para poder ser clasificado en c_i. Tenga en cuenta que esto nos permite ver la construcción de un clasificador para el conjunto de categorías $C = \{c_1, \dots, c_m\}$ como m tareas independientes de construir un clasificador para cada categoría $c_i \in C$, luego cada uno de estos clasificadores será una regla que permitirá decidir si el documento d_j debe clasificarse en la categoría c_i o no.

Como se mencionó anteriormente, el enfoque de aprendizaje automático se basa en la existencia de un corpus inicial $Co = \{d_1, \dots, d_s\}$ de documentos anteriormente clasificados bajo el mismo conjunto de categorías $C = \{c_1, \dots, c_m\}$ con el que el clasificador debe operar. Esto significa que el corpus viene con una matriz de decisión correcta [7].

	Conjunto de entrenamiento				Conjunto de prueba			
	$\bar{d}_1$	⋯	⋯	$\bar{d}_g$	$\bar{d}_{g+1}$	⋯	⋯	$\bar{d}_s$
c_1	ca_{11}	⋯	⋯	ca_{1g}	$ca_{1(g+1)}$	⋯	⋯	ca_{1s}
⋯	⋯	⋯	⋯	⋯	⋯	⋯	⋯	⋯
c_i	ca_{i1}	⋯	⋯	ca_{ig}	$ca_{i(g+1)}$	⋯	⋯	ca_{is}
⋯	⋯	⋯	⋯	⋯	⋯	⋯	⋯	⋯
c_m	ca_{m1}	⋯	⋯	ca_{mg}	$ca_{m(g+1)}$	⋯	⋯	ca_{ms}

Un valor de 1 para ca_{ij} se interpreta como una indicación del experto de presentar d_j bajo c_i, mientras que un valor de 0 se interpreta como una indicación del experto de no presentar d_j bajo c_i. Un documento d_j se refiere a menudo como un ejemplo positivo de c_i si $ca_{ij} = 1$, y será un ejemplo negativo de c_i si $ca_{ij} = 0$.

Para fines de evaluación, en la primera etapa de la construcción del clasificador, el corpus inicial se divide típicamente en dos conjuntos, no necesariamente de igual tamaño:

- Un conjunto de entrenamiento $Tr = \{\bar{d}_1, \ldots, \bar{d}_g\}$. Este es el conjunto de documentos de ejemplo para la observación de las características con las cuales los clasificadores para las distintas categorías son inducidos;
- Un conjunto de pruebas $Te = \{\bar{d}_{g+1}, \ldots, \bar{d}_s\}$. Este conjunto se utiliza para el propósito de probar la efectividad de los clasificadores inducidos. Cada documento de Te será comprobado con los clasificadores, y las decisiones que el clasificador tome en comparación con las decisiones de los expertos, crearán una medida de la efectividad de la clasificación, la cual se basa en la frecuencia con la que los valores de las a_{ij}'s pueden obtenerse a través de los clasificadores de acuerdo a los valores de las ca_{ij}'s proporcionados por los expertos.

Las fases de la tarea de clasificación son:

- Fase de indexación y reducción de la dimensionalidad
- Fase de la construcción del clasificador
- Fase de la evaluación del clasificador.

4.1 Indexación y reducción de la dimensionalidad

Cada documento (ya sea perteneciente al corpus inicial, o para ser clasificados en la fase de operación del sistema) suele ser representado mediante un vector de n términos. Una forma simple de representar el documento es mediante las palabras que lo componen. En este caso se refiere a menudo al enfoque de la representación del documento como *bag of words (BOW)*.

En la clasificación de documentos la alta dimensionalidad del espacio de términos, es decir, el hecho de que el número r de términos que aparecen al menos una vez en el corpus Co es alto, puede ser problemática. Debido a esto, las técnicas de reducción de dimensionalidad (DR) se emplean a menudo y su efecto es reducir la dimensionalidad del espacio del vector de r a r' siendo r'<< r.

La reducción de dimensionalidad es también beneficiosa, ya que tiende a reducir el problema de sobreaprendizaje (*overfitting)*, es decir, el fenómeno por el cual un clasificador se sintoniza también al contingente, no sólo con las características necesarias de los datos de entrenamiento. Clasificadores con *overfitting* de datos de entrenamiento tienden a ser muy buenos en la clasificación de los datos con los que han sido entrenados, pero son notablemente malos en la clasificación de otros datos.

Varias funciones de DR, ya sea a partir de la teoría de la información o de la literatura del álgebra lineal, se han propuesto,

y sus méritos relativos se han probado experimentalmente por la evaluación de la variación en la efectividad de la clasificación.

Hay dos maneras muy distintas de DR, dependiendo de si la tarea se acerca a nivel local, es decir, para cada categoría individual, en forma aislada de las otras o globalmente:

- La reducción de dimensionalidad local: se eligen las características por cada categoría.
- La reducción de dimensionalidad global: se eligen las mismas características para todas las categorías.

Esta distinción no repercute en el tipo de técnica elegida para la DR, ya que la mayoría de las técnicas de DR se pueden utilizar (y se han utilizado) ya sea para la reducción de dimensionalidad local o global.

Una segunda distinción ortogonal puede redactarse en términos del tipo de características que son escogidas:

- Reducción de dimensionalidad por selección de características: las características elegidas son un subconjunto de las características originales r, dichas características son las más representativas;
- Reducción de dimensionalidad por extracción de características: las características elegidas no son un subconjunto de las características originales r. Por lo general, las elegidas no son homogéneas con las características originales (por ejemplo, si las características originales son palabras, las características elegidas no pueden ser palabras en absoluto), sino que se obtienen mediante combinaciones o transformaciones de las originales.

  Obviamente, las dos formas diferentes de hacer las DR se abordan mediante técnicas muy distintas.

4.2 Construcción del clasificador

El problema de la construcción inductiva de un clasificador de texto ha sido abordado en una variedad de maneras diferentes:

1. Definición de una función $CSV_i: D \rightarrow [0,1]$ que, dado un documento d, devuelve un *valor del estado de categorización* para ello, es decir, un número entre 0 y 1 que, en términos generales, representa la evidencia por el hecho de que d se clasifica en c_i. La función CSV toma diferentes significados de acuerdo con los clasificadores diferentes: por ejemplo, con el criterio "Naive Bayes" que se expone en la Sección 6.2 $CSV(d)$ es la probabilidad de que d pertenence a c_i.

2. Definición de un *umbral* (*threshold*) τ_i, de tal manera que $CSV_i(d) \geq \tau_i$ se interpreta como una decisión para clasificar d bajo c_i, mientras $CSV_i(d) < \tau_i$ se interpreta como una decisión de no categorizar d bajo c_i. Un caso particular se produce cuando el clasificador proporciona ya un juicio binario, es decir $CSV_i: D \rightarrow \{0,1\}$. En este caso, el umbral es trivialmente cualquier valor en el intervalo (0,1) abierto.

4.3 Cuestiones de evaluación

La evaluación de los clasificadores de documentos se realiza típicamente de forma experimental, en vez de analíticamente. La evaluación experimental de los clasificadores, en lugar de concentrarse en las cuestiones de eficiencia, por lo general trata de evaluar la eficacia de un clasificador, es decir, su capacidad de tomar las decisiones correctas de categorización. Las principales razones de esta tendencia son las siguientes:

- La eficiencia es un concepto dependiente de la tecnología de hardware / software utilizada. Una vez que la tecnología evoluciona, los resultados de los experimentos encaminados a establecer la eficiencia ya no son válidas. Esto no sucede con la eficacia, ya que cualquier experimento destinado a medir la efectividad puede ser replicado, con idénticos resultados, ya sea con una plataforma diferente o futura de hw / sw;
- La eficacia es realmente una medida de la forma en que el sistema es bueno al abordar la noción central de la clasificación, el de la relevancia de un documento a una categoría.

Colecciones de ensayo

A fin de que los resultados experimentales en dos clasificadores diferentes puedan ser directamente comparables, los experimentos deben realizarse en las siguientes condiciones:

1. La misma colección (es decir, los mismos documentos y las mismas categorías) se utiliza para los clasificadores;
2. La misma opción del conjunto de entrenamiento y de prueba se utiliza para ambos clasificadores;
3. La medida de efectividad es la misma para ambos clasificadores.

Medidas de la eficacia de la clasificación

Precision (Pr) y recall (Re)

La precisión $c_i(Pr_i)$ se define como la probabilidad condicional $P(ca_{ix} = 1|a_{ix} = 1)$, es decir, como la probabilidad de que si un documento d_x aleatorio se clasifica en c_i, esta decisión es correcta. Análogamente, el *recall* $c_i(Re_i)$ se define como la probabilidad condicional $P(a_{ix} = 1|ca_{ix} = 1)$, es decir, la probabilidad de que, si

un documento d_x aleatorio debe ser clasificado en c_i, esta decisión se toma. Estos valores relativos de categoría pueden ser promediados, para obtener Pr y Re, es decir, valores globales para el conjunto completo de categorías. Un préstamo de la terminología de la lógica, Pr puede ser visto como el "grado de solidez" del clasificador dado el conjunto de categorías C, mientras que Re se puede ver como su "grado de completitud".

Estas probabilidades pueden estimarse en términos de la *tabla de contingencia* (denominada *confusion matrix* en Weka) para la categoría c_i en un conjunto de prueba dado (ver Tabla 4.1). Aquí, FP_i (falsos positivos c_i) es el número de documentos de la unidad de prueba que han sido incorrectamente clasificada en c_i; TN_i (verdaderos negativos c_i), TP_i (verdaderos positivos c_i) y el FN_i (falsos negativos c_i) definen en consecuencia. *Precision* c_i y el *recall* c_i tanto, puede estimarse como:

Categoria c_i		opiniones de expertos	
		YES	**NO**
juicios del clasificador	**YES**	TP_i	FP_i
	NO	FN_i	TN_i

Tabla 4.1: Tabla de contingencia para la categoría c_i.

$$Pr_i \overset{est}{=} \frac{TP_i}{TP_i + FP_i}$$

$$Re_i \overset{est}{=} \frac{TP_i}{TP_i + FN_i}$$

Para la obtención de las estimaciones de *precision* y *recall* en relación con el conjunto de categorías general, dos métodos diferentes pueden ser adoptados:

Conjunto de Categorías		opiniones de expertos	
		YES	**NO**
juicios del clasificador	**YES**	$TP = \sum_{i=1}^{m} TP_i$	$FP = \sum_{i=1}^{m} FP_i$
	NO	$FN = \sum_{i=1}^{m} FN_i$	$TN = \sum_{i=1}^{m} TN_i$

Tabla 4.2: Tabla global de contingencias.

- *microaveraging*: la *precision* y el *recall* se obtienen sumando globalmente todas las decisiones individuales, es decir:

$$Pr^{\mu} est = \frac{TP}{TP + FP} = \frac{\sum_{i=1}^{m} TP_i}{\sum_{i=1}^{m} (TP_i + FP_i)}$$

$$Re^{\mu} est = \frac{TP}{TP + FN} = \frac{\sum_{i=1}^{m} TP_i}{\sum_{i=1}^{m} (TP_i + FN_i)}$$

  donde el superíndice "μ" representa el *microaveraging*. Para ello, se utiliza la tabla "global" de contingencia (Tabla 4.2), que se obtiene sumando todas las tablas de contingencia específicas de categoría.

- *macroaveraging*: la *precision* y el *recall* son evaluados primero de forma "local" para cada categoría y, a continuación, de forma "global" haciendo un promedio sobre los resultados de las diferentes categorías, es decir:

$$Pr^{M} est = \frac{\sum_{i=1}^{m} Pr_i}{m}$$

$$Re^{M} est = \frac{\sum_{i=1}^{m} Re_i}{m}$$

  donde el superíndice "M" representa el *macroaveraging*.

Es importante reconocer que estos dos métodos pueden dar resultados muy diferentes, especialmente si las diferentes categorías se conforman desigualmente: por ejemplo, si el clasificador se desempeña bien en categorías con un pequeño número de casos positivos de prueba, su eficacia será probablemente mejor según el *macroaveraging* que con el *microaveraging*. No hay acuerdo entre los autores sobre cuál es mejor.

Medidas combinadas

Ni la precisión ni el *recall* tienen sentido en aislamiento. Es bien conocido que a mayores niveles de precisión puede obtenerse un *recall* bajo.

Un clasificador por lo tanto se debe medir por medio de una medida "combinada", la eficacia se logrará combinando Pr y Re. La medida más utilizada en el ámbito de la clasificación automática de textos es el *F-measure*:

$$F - Measure = \frac{2 \times Pr \times Re}{(Pr + Re)}$$

5 RECOPILACIÓN DE DOCUMENTOS

Es necesario un corpus de textos en quechua con las categorías asignadas, que en este caso serán: quechua cusqueño y quechua no cusqueño, para el entrenamiento y prueba de nuestro clasificador automático de textos, como se ha descrito anteriormente.

El corpus conseguido para poder desarrollar el presente trabajo, proviene de las siguientes fuentes:

- Universidad Nacional Intercultural de la Amazonía
- Asociación Interdenominacional para el Desarrollo Integral de Apurímac
- Instituto Superior "La Salle" -PROYECTO CRAM II
- Instituto Superior Pedagógico Público Túpac Amaru.
- Academia de la Lengua Quechua filial Apurímac
- INKAWASI – Lambayeque
- SIL International y Universidad Ricardo Palma
- Chirapaq, Centro de Culturas Indígenas el Perú
- Ministerio de Educación del Perú
- BibliotecaNacional del Pe(www.lengamer.org)

La tabla 5.1 muestra un resumen de los datos del corpus recopilado, como el número total de documentos recopilados, número total de palabras y número de dialectos del corpus.

Número de documentos			85
Nro. de palabras	*Palabras originales*	*Total*	618,913
		Diferentes	132,012
	Palabras del alfabeto quechua	*Total*	463,511
		Diferentes	28,240
Número de dialectos			8

Tabla 5.1: Datos del corpus de texto quechua.

La mayor parte de textos recopilados son de tipo cuento (34) y textos escolares de educación primaria (25). Se puede apreciar con mayor detalle las características del corpus en el Anexo 1.

La tarea de recopilación de documentos escritos en quechua para conformar el corpus necesario para la construcción del clasificador automático, fue ardua y meticulosa, esto debido a la escasa existencia de documentos escritos en este idioma y fue aún más difícil hallarlos en formato electrónico tratable para los propósitos de clasificación que buscamos; por ello se tuvieron que realizar previamente las tareas de digitalización y pre-proceso de textos, que se detallan a continuación:

5.1 Digitalización de textos

Debido a que una gran parte de los textos obtenidos estaban en papel, se tuvo que escanear primero dichos documentos y luego procesarlos con ayuda de un OCR, para posteriormente hacer una revisión y corrección manual del contenido de cada archivo y finalmente obtener un archivo de texto por cada documento. También se debió de hacer una revisión y corrección manual de los documentos encontrados en formato digital, debido a que éstos no estaban en un formato digital de fácil tratamiento para su paso a archivos de texto, porque eran documentos escritos en máquina de escribir o estaban mal escaneados.

El corpus obtenido al finalizar la tarea de digitalización, se compone de 85 documentos clasificados en 2 categorías: quechua cusqueño y quechua no cusqueño.

5.2 Pre-procesamiento de textos

Una vez obtenido el corpus en archivos de texto de posible manejo se implementaron programas para la limpieza de:

- *Caracteres extraños*, como números, caracteres especiales, signos de puntuación, como por ejemplo: - _ " ! ¡ ? ¿, entre otros.

  El resumen de las características de este corpus digitalizado y libre de caracteres extraños se puede apreciar en la tabla 5.2.

Categoría	*Nro. de documentos*	*Nro. de palabras*	*Nro. de palabras diferentes*
Quechua Cusqueño	37	583,532	117,181
Quechua no Cusqueño	48	35,381	14,840
Total	85	618,913	132,021

Tabla 5.2: Detalle del corpus de texto escrito en quechua.

Se observa en la tabla 5.2 que el número de documentos en quechua cusqueño es menor al número de documentos en quechua no cusqueño, sin embargo el número total de palabras obtenidas de los documentos en quechua cusqueño supera ampliamente al número de palabras obtenidas de los documentos escritos en quechua no cusqueño, esta gran diferencia se explica debido a la presencia de la Biblia entre los documentos en quechua cusqueño, la cual contiene 453,632 palabras.

Al corpus obtenido luego de este pre-proceso se denominará *bag of words* (BOW).

- *Palabras con letras que no pertenecen al alfabeto quechua,* también se limpiaron aquellas palabras que contenían las

letras "b", "d", "g", "v", "x", "z". Se eliminaron estas palabras con el fin de evitar tener palabras castellanizadas que son iguales en todos los dialectos.

- *Palabras con menos frecuencia,* se eliminaron también las palabras que aparecían menos de tres veces en el corpus, dado que muchas de ellas provienen de errores tipográficos o son palabras de tan poco uso que no resultan discriminatorias para la clasificación que nos ocupa.
  A continuación se presenta una tabla resumen del corpus luego del pre-proceso de eliminación de palabras con caracteres extraños, palabras con letras no pertenecientes al alfabeto quechua y palabras menos frecuentes.

Categoría	*Nro. de documentos*	*Nro. de palabras*	*Nro. de palabras diferentes*
Quechua Cusqueño	37	439,494	23,214
Quechua no Cusqueño	48	24,017	5,026
Total	85	463,511	28,240

Tabla 5.3: Corpus después de la eliminación del pre-procesado.

Al corpus obtenido luego de eliminar palabras no pertenecientes al alfabeto quechua y palabras de menor frecuencia de aparición, se le denominará *bag of words* quechua (BOWQ).

- *Stop Word List.* Se creó una lista de palabras funcionales para eliminarlas del corpus. Para ello se tomó inicialmente las 200 palabras más frecuentes, las que se repetían en casi todos los documentos y que se consideran palabras funcionales. Se tuvo

que retocar esta la lista debido a que aparecieron palabras muy frecuentes pero con información semántica mezcladas con las palabras funcionales. También se decidió disminuir el *stop word list* de 200 palabras (SL200) a uno de 150 palabras (SL150), luego a uno de 100 palabras (SL100) y finalmente se probó con uno de 90 palabras (SL90). Se observó mejoras muy considerables en los resultados experimentales con el *stop word list* de 100 palabras, resultados que ya se verán en la sección 7. A continuación se presenta la tabla 5.4 donde se resume las diferencias en el corpus aplicando los 4 *stop word list.*

Categoría	*Nro. de documentos*	*SL200*	*SL150*	*SL100*	*SL90*
Quechua Cusqueño	37	314,453	324,880	353,186	358,139
Quechua no Cusqueño	48	16,664	17,459	19,691	20,076
Total	85	331,117	342,339	372,877	378,215

Tabla 5.4: Número total de palabras del corpus con los 4 *stop word list.*

Categoría	*Nro. de documentos*	*SL200*	*SL150*	*SL100*	*SL90*
Quechua Cusqueño	37	23,016	23,065	23,115	23,125
Quechua no Cusqueño	48	4,830	4,879	4,928	4,938
Total	85	27,846	27,944	28,043	28,063

Tabla 5.5: Número de palabras diferentes del corpus con los 4 *stop word list.*

- *Lematización,* se desarrollaron programas para obtener una lista de las raíces correspondientes a todas y cada una de las palabras del corpus, se trabajó la lematización sobre dos corpus; uno luego de la validación de las palabras con el

alfabeto quechua y eliminación de palabras menos frecuentes (BOLQ) y otro luego del pre-procesado con el *stop word list* de 100 palabras (LSL100). Estos programas se basaron en el analizador morfológico de Annette Ríos [2]. Básicamente se fue retirando uno a uno los sufijos tomando en consideración el orden de los sufijos que se detalla en este trabajo, son aproximadamente 106 diferentes sufijos, sin embargo otros trabajos como [3], [12] y [21] consideran otros sufijos adicionales. Hasta donde se tiene conocimiento todos los sufijos quechua son iguales para todos los dialectos excepto el caso de la conjugación de los verbos de la primera persona en singular como se puede ver en [10], por no tener más conocimiento al respecto, no se considerarán estos sufijos en este trabajo. A continuación en la tabla 5.6 se muestran estos sufijos quechua, a qué tipo de raíces se pueden aplicar cada uno de ellos y la posición probable que pueden ocupar después de la raíz.

Posición después de la raíz	**Denominación del sufijo**	**Tipo de raíz**	
		Nominal	**Verbal**
1	Sufijos de nominalización		Si
1	Sufijos de verbalización	Sí	
2	Derivación	Sí	Si
3	Posesión	Sí	
3	Aspecto		Si
4	Tiempo		Si
4	Caso	Sí	
5	Persona		Sí
6	Modalidad		Sí
7	Sufijos ambivalentes	Sí	Sí

Tabla 5.6: Listado general de sufijos quechua.

Dentro de los sufijos de verbalización se distinguen 7 tipos, los cuales sólo podrán ser aplicados a las raíces de tipo nominal, como se puede observar en la tabla 5.7.

Denominación del sufijo	**Sufijo**
Autotransformativo	*Lli*
Desiderativo	*Naya*
Reubicativo	*Na*
Factivo	*Cha*
Caracterización	*Raya*
Transformativo	*Ya*
Simulativo	*kacha/~ykacha*

Tabla5.7: Sufijos de verbalización.

Para el caso de sufijos de nominalización, aplicables sólo a raíces de tipo verbal se distinguen 7 tipos de sufijos que se pueden observar en la tabla 5.8.

Denominación del sufijo	**Sufijo**
Infinitivo	*Y*
Caracterización	*ti/~li/~lu*
Obligación	*Na*
Agentivo	*Q*
Posicional	*Mpa*
Perfecto	*Sqa*
Mismo tema	*spa/shpa*
Mismo tema simultáneo	*sti/~stin*
Diferente tema	*pti/qti*

Tabla 5.8: Sufijos de nominalización.

La tabla 5.9 muestra una relación de sufijos así como la descripción de estos, aplicables a raíces nominales, como se puede observar a continuación:

Descripción			Sufijo
Derivación	Derivación de la raíz	Diminutivo	*cha*
		Aumentativo	*chika* *karay* *chaq* *su*
		Similitud	*niraq* *rikuq*
		Caracterización	*ti* *li* *liku* *yli* *lu*
		Posicional	*mpa*
Posesión	Derivación de posesor	Posesor múltiple	*sapa*
		Posesor ausente	*yuq* *nnaq*
Caso	Numero	Plural	*kuna*
	Caso 1	Inclusivo	*ntin*
		Interasociativo	*pura*
		Distributivo	*nka*
		Terminativo	*kama*
		Aproximativo	*niq*
		Locativo	*pi*
		Benefactivo	*paq*
	Caso 2	Acusativo	*ta*
		Genitivo	*pa*
		Prolocativo	*nta*
		Ablativo	*manta*
		dativo/ilativo	*man*
	Caso 3	Distributivo	*kama*
		instrumental/conectivo	*wan*
		Causa	*rayku*
		Asociativo	*puwan*

Tabla 5.9: Sufijos aplicables sólo a raíces nominales.

En la tabla 5.10 se muestra una relación de sufijos con su respectiva descripción, aplicables a raíces verbales.

Descripción			Sufijo
Derivación	Derivación de la raíz	Rememorativo	*ymana*
		Simulativo	*tiya*
		Desesperativo	*pasa*
		Intencional	*rpari*
		Urgencia	*rqu*
		verbal diminutivo, infantilismo	*cha*
		Repetitivo	*pa*
		Incoativo	*ri*
		Autotransformativo	*lli*
		Interruptivo, frecuentativo	*ykacha/kacha*
		Continuidad	*nya/miya*
		Afectivo	*yku*
		Multi-repetitivo	*paya*
	Sufijos de cambio de valencia	Asistencia	*ysi*
		Desiderativo	*naya*
		Reciproco	*na*
		Causativo	*chi*
		Perdurativo	*raya*
	Direccionales & Reflexivo	Reflexivo, intensificador	*ku*
		Regresivo, interpersonal	*pu*
		Cislocativo, translocativo	*mu*
	Objeto	Objeto 1ra Persona	*wa*
		Objeto 2da Persona	*su*
Aspecto		Progresivo	*sha*
Tiempo		Pasado neutral	*rqa*
		Pasado narrativo	*sqa*
Modalidad		Potencial	*man*
Sufijos ambivalentes	(igual al de los nominales)		

Tabla 5.10: Sufijos aplicables sólo a raíces verbales.

En la tabla 5.11 se listan los denominados sufijos ambivalentes por ser aplicables tanto a raíces verbales como nominales.

Descripción		Sufijo
Sufijos ambivalentes	Similitud	*hina*
	Certeza	*puni*
	Aditivo	*pas*
	Continuativo	*raq*
	Descontinuativo	*ña*
	Conectivo/Interrogativo	*taq*
	Negación/Interrogativo	*chu*
	Evidencia directa	*mi/m*
	Evidencia indirecta	*si/s*
	Supuesto	*cha/ch*
	Tópico	*qa*
	Tópico en preguntas	*ri*
	Dubitativo	*suna*
	Resignación, Implícito	*iki*
	Enfático	*ya*
	Evidencia directa – Enfático	*ma*
	Evidencia indirecta – Enfático	*sa*
	Supuesto – Enfático	*cha*

Tabla 5.11: Sufijos ambivalentes aplicables a raíces verbales y nominales.

En la tabla 5.12 se podrá observar el número de diferentes lemas resultantes luego de aplicar la lematización sobre los corpus BOWQ y SL100 descritos anteriormente.

Se observa que del número total de palabras distintas del corpus inicial BOWQ es de 28,240, y luego de realizar el proceso de la lematización quedaron 5,474 lemas correspondientes a dichas palabras.

Del mismo modo se observa que en el caso del corpus SL100 el número total de palabras distintas es de 28,043, y luego de realizar la lematización de este quedaron 5,471 lemas.

Categoría	*Nro. de documentos*	*BOLQ*	*LSL100*
Quechua Cusqueño	37	4,357	4,356
Quechua no Cusqueño	48	1,117	1,115
Total	85	5,474	5,471

Tabla 5.12: Número de lemas diferentes sobre los corpus BOWQ y SL100.

6 EXPERIMENTACIÓN

6.1 Herramienta

Para la construcción de los diferentes clasificadores se ha utilizado la plataforma de software para aprendizaje automático y minería de datos Weka (*Waikato Environment for Knowledge Analysis* - Entorno para Análisis del Conocimiento de la Universidad de Waikato) que está escrito en Java. Weka además es un software libre, distribuido bajo licencia GNU-GPL [8]. Esta plataforma ofrece varias cualidades para el desarrollo de este trabajo, como una interfaz gráfica que hace más sencillo su uso, varias técnicas para el procesamiento de datos, además de implementar los principales algoritmos de clasificación.

Para poder emplear esta plataforma, se han creado ficheros arff para cada uno de los corpus antes descritos con la ayuda de otros programas. Un ejemplo de la cabecera y parte del contenido de estos ficheros se observa en la figura 6.1.

```
@relation tcq

@attribute doc string
@attribute cat {0,1}

@data
'atskaq nunakunapa kallpanqa imaaykatapis ahallam ruran huk nunapam kapurqan
qanchis tsurinkuna papanin mamanin imeypis keynowmi niyaq felis keyta munarqa
kayaneyki unidum tseynow papaninkuna niyaptin tsurinkuna mana wiyakuyarqatsu
allapa dejado kayaq mana nunka yanapanakuyaqtsu kasunakuyaqtsu imeypis
maqanakullar kakullaq tseyta rikarnam limpu llakishqa teytankuna llapanta
eylliykuq willapananpaq tsey eyllukariyaptinnam mañashqa huk llanu
shukshukunata kada tsurinta tseytanam papaninkuna eyllurkur alli bwenu
watashqa tsey shukshukuna watashqata tsaratsirqan tsurinkunata keynow nishpa
ma keyta pakirayami tseynam wamrankunaqa pakita munayashqa pero mana
pwedeyáshqatsu qanchis llanu shukshu watashqata llapan huntupis mana pakita
pwedeyashqatsu tseynam papaninkuna paskarinaq qanchis shúkshuta tsurinkunata
tsayaratsirqan ma pakiyay kaynaw nishpa ma kanan pakiyay tséytaqa kada
wamrankuna hukllapa pakiriyarqan tseynam papaninkuna keynow nirqan yarpayey
keyta llapan kaweynikikunachow tseyyaq mamaninkuna upallalla keykanqanta
meqeykikunapis hapalleekikuna kayanqeyki ora kayanki key llanu shukshunowmi
fasil pakiripaq llapeykitsun huntu kayanki tseyqa kayanki huk alli puqushqa i
chukru qirunowmi ' 0
'huk punchawsi maria michisharan qatapi wakanta chaysi joseyqa qonqaylla
qepanmanta hapispa mariata manchachiran hinaspas maria khaynata nisqa yaw
joséchay maqta imanaqtinmi manchachiwanki nispa chaysi joseyqa nisqa munasqay
warmachallay anchatapunin munakuyki sonqoymi pun pun ninraq qanmanta soqta
killañan compromisunchisqa kashan casarakullasunñaya nispa chaysi mariaqa puka
uyallantinñan nisqa ari noqapas munakuykin hinallataqmi casarakuyta munani
qanwan nispa hinaqtinsi ishkayninku rimasqankuta taytamankuwan parlanankupaq
wakakunata qaterikuspankus kutinpushasqanku kusisqallaña mariaq wasinman
chayarunanku kashaqtinsi papanqa phiñallaña lloqsimusqa khaynata nispa yaw
```

Figura 6.1: Cabecera de un archivo arff.

Como se observa el fichero tiene 2 atributos:

- En el atributo *doc*, que es de tipo *string*, se almacenará la cadena de palabras de todos y cada uno de los diferentes archivos de texto correspondientes a cada documento del corpus.
- El atributo *cat* almacenará la categoría correspondiente a cada documento: 1 si el documento pertenece al dialecto quechua cusqueño y 0 si el documento pertenece al dialecto quechua no cusqueño.

Posteriormente se aplicará el filtro *StringToWordVector* al atributo *doc*, debido a que el tipo de datos *String* es de difícil tratamiento en los algoritmos de aprendizaje automático porque normalmente estos algoritmos trabajan con datos numéricos o binarios, por ello es necesario transformar los documentos a vectores. Para ello Weka proporciona diversos filtros para este tipo de transformaciones.

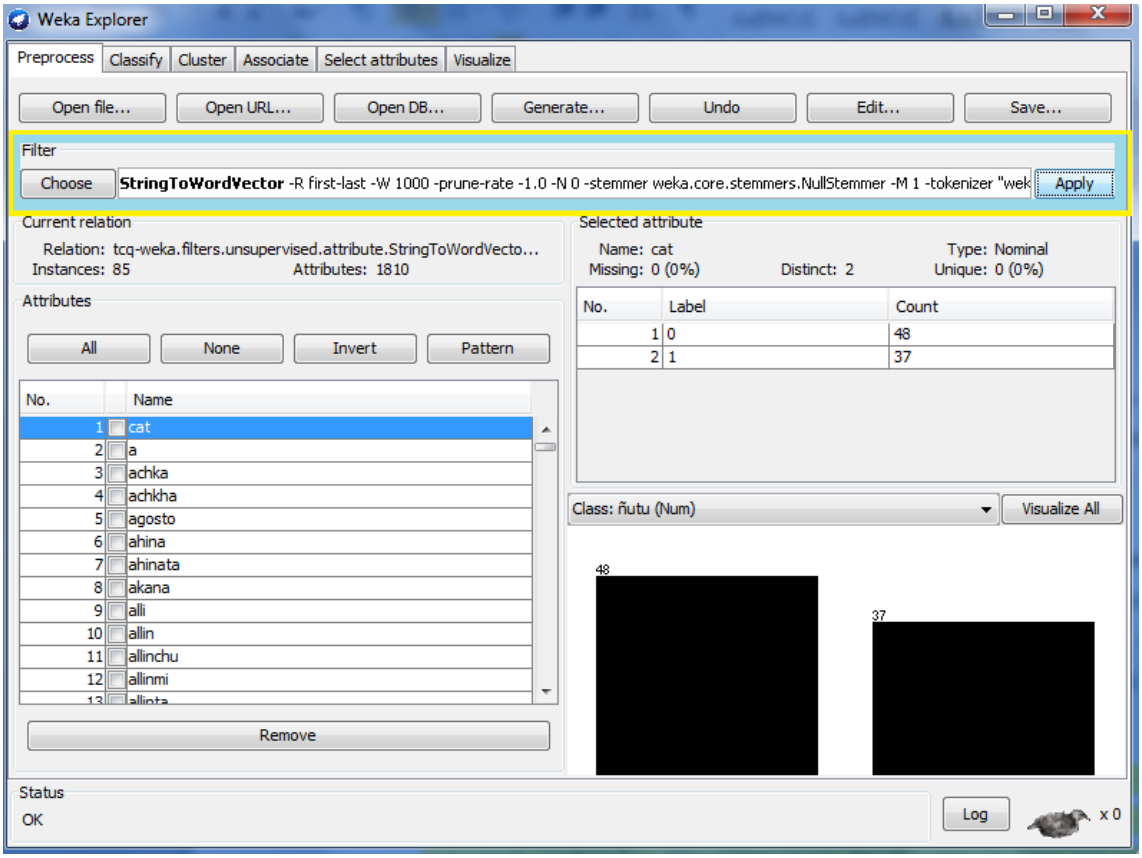

Figura 6.2: Aplicación del filtro *StringToWordVector* a un archivo arff.

Este filtro convierte cada una de las palabras del documento en un atributo al que se le puede asignar un valor, como puede verse en 6.2 cada palabra aparece como un nuevo atributo.

El *StringToWordVector* es uno de los filtros que proporciona Weka, que convierte atributos de tipo *string* en un conjunto de atributos, los cuales representan la ocurrencia de cada una de las palabras, en el corpus.

6.2 Algoritmos

Este trabajo es una primera aproximación de la clasificación de textos escritos en quechua, por ello, lo que se pretende es probar

los diferentes tipos de algoritmos para ver cuál de ellos proporciona mejores resultados.

Se han elegido los siguientes algoritmos debido a su frecuente uso en la literatura sobre clasificación de textos:

Naive Bayes

Este clasificador se basa en la teoría de Bayes suponiendo que hay una independencia entre los atributos de los individuos del modelo. Se calculan las distribuciones de probabilidad de cada clase para establecer la relación entre los atributos y la clase (variable dependiente).

Árboles de decisión J48

Se crea un árbol de decisión tomando para cada nodo del árbol el atributo, no utilizado, cuya entropía es menor, haciendo que el nodo aporte la mayor cantidad de información posible.

Reglas PART

Genera una lista de decisión sin restricciones usando el procedimiento de "divide y vencerás". Además construye un árbol de decisión parcial para obtener una regla. Es conveniente la poda para evitar el *overfitting*. Para poder podar una rama (una regla) es necesario que todas sus implicaciones sean conocidas.

Reglas JRip

JRip (*Repeated Incremental Pruning*) Este es un algoritmo que genera un listado de reglas obtenidas básicamente a partir de listas de decisión. Funciona de modo similar a RIPPER (*Repeated Incremental Pruning to Produce Error Reduction*). Este es el formalismo para la creación de reglas de JRip, consiste en hacer una lista ordenada de reglas conjuntivas y evaluarlas en orden para encontrar la primera regla que se cumple sobre el ejemplo a clasificar. Una vez encontrada dicha regla se ha encontrado la regla más eficiente para ese ejemplo.

Funciones SMO

SMO (*Sequential Minimal Optimization*) es un algoritmo iterativo para resolver el problema de optimización para entrenar SVM (*margin support vector machine*). SMO divide este problema en una serie de sub-problemas, que luego son resueltos analíticamente.

Por otro lado, se evaluará la calidad del clasificador usando la opción de entrenamiento del *cross validation* (validación cruzada). Se elige esta opción debido a que el corpus con el que realizaremos los experimentos es pequeño.

Se probarán 2 opciones, 5-*fold cross validation* (CV5) y 10-*fold cross validation* (CV10).

En la Figura 6.3 se muestra un ejemplo de la técnica 4-*fold cross validation*, en la que se parte el conjunto de datos inicial en 4 subconjuntos. Para testear cada una de las partes se entrena el clasificador con las partes restantes y se hace una media de las tasas de acierto de cada test realizado.

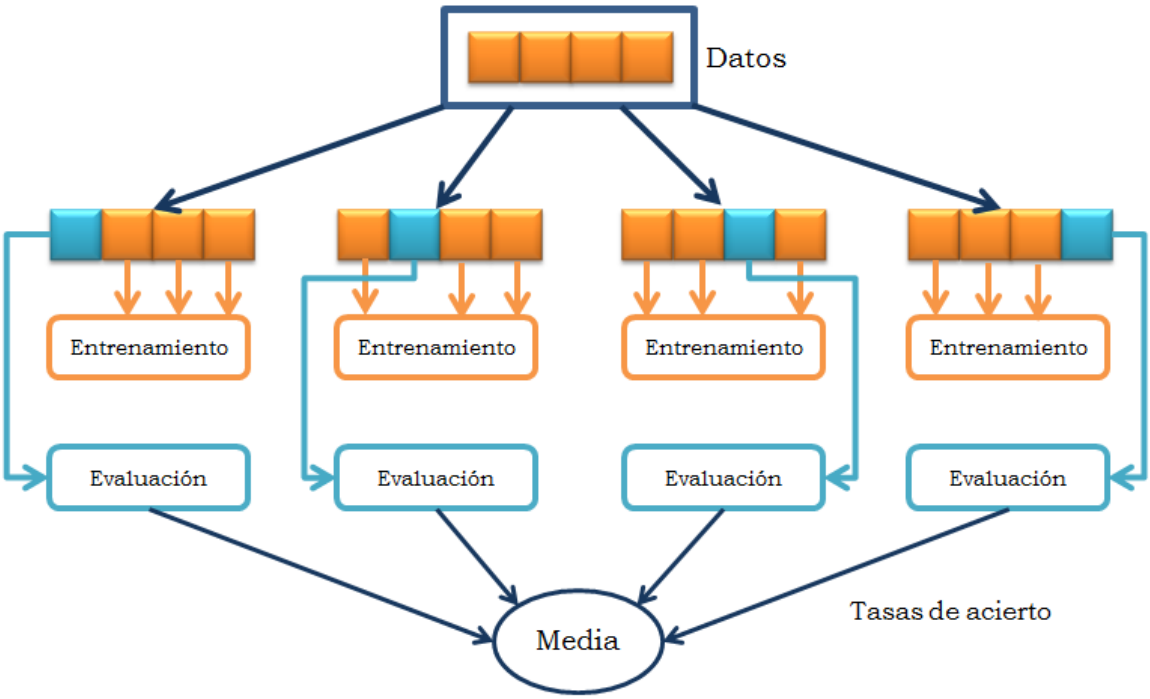

Figura 6.3: Esquema k-*fold Cross Validation*, con k=4.

6.3 Atributos para la clasificación

La representación de los documentos es una cuestión importante a considerar, por ello la selección de los atributos con los que se representarán las características de cada texto es una tarea trascendental, porque de esto dependerá el éxito del clasificador.

Se ha visto en la bibliografía que la mayoría de los estudios efectuados, realizan una clasificación basada en las palabras que componen el texto a clasificar, *bag of words*; sin embargo también existen estudios que tratan de analizar el impacto que puede producir la utilización de otro tipo de características para representar el texto que se pretende categorizar, como pueden ser los lemas y N-gramas.

La experimentación se realizó con atributos tomados individualmente en un principio y después combinándolos.

Palabras

Consideraremos palabras a cada uno de los segmentos limitados por delimitadores, en nuestro caso serán espacios en blanco y signos de puntuación (. , ; : ¿ ¡ ...).

a) Inicialmente se toman todas las palabras que aparecen en todos y cada uno de los documentos originales conseguidos para una primera fase lo que habitualmente se denomina *bag of words* BOW. El corpus BOW inicial de este trabajo consta de 132,021 palabras distintas.
b) Luego se consideran sólo las correspondientes al alfabeto quechua y que aparezcan por lo menos 3 veces en todo el corpus BOWQ. Hasta este punto se tienen 28,240 palabras distintas.
c) Finalmente se hará una limpieza de esta última considerando el *stop word list* de 200 palabras (SL200), el *stop word list* de 150 palabras (SL150), el *stop word list* de 100 palabras (SL100) y el *stop word list* de 90 palabras (SL90).

En estos conjuntos de características se toma la forma completa de la palabra.

Ejemplos:

El texto original como corpus BOW (descrito en el ítem (a) anterior) quedaría así:

> *Aristotelesmi Organonpi k'uskirqan logika nisqata kikin kayninpipas ch'uya kayninpi hinallataq pachawan tupaqninkunapipas kunan teoría del conocimiento nisqapi hinallataq gnoseología nisqapipas*

Luego de aplicar lo considerado en (b) quedaría:

Aristotelesmi k'uskirqan nisqata kikin kayninpipas ch'uya kayninpi hinallataq pachawan tupaqninkunapipas kunan teoría conocimiento nisqapi hinallataq nisqapipas

Finalmente luego de aplicar lo considerado en (c) quedaría:

k'uskirqan nisqata kayninpipas ch'uya kayninpi pachawan nisqapi nisqapipas

Lemas

Al ser el quechua una lengua aglutinante es importante considerar como una característica los lemas, porque estos aglutinan la información de varias formas. Además resulta interesante experimentar la diferencia que pueda tener la aplicación de lemas. Por un lado se espera aglutinar la información si bien para el euskera no fue muy determinante. Son aproximadamente 106 sufijos quechua los considerados para esta tarea.

Se empleará la lematización sobre los siguientes corpus:

a) Corpus obtenido luego de la limpieza de caracteres extraños, palabras no pertenecientes al alfabeto quechua y palabras con menor frecuencia de aparición, descrito con más detalle en a y b del apartado anterior. Al corpus obtenido así lo denominaremos BOLQ.
b) También aplicaremos lematización al corpus sobre el cual aplicamos *stop word list,* pero en vez de probar todas las combinaciones posibles nos centraremos en aquel que mejor rendimiento obtenga.

N-gramas

Emplearemos también como técnica de representación de las características de los textos, los N-gramas, porque como se mencionó en el capítulo 2, existen antecedentes de buenos resultados en la clasificación automática de textos por lenguaje, además de ofrecer las siguientes ventajas:

- No es necesario un pre-procesamiento lingüístico
- Es independiente del idioma
- Menos dispersión de datos

Para los experimentos emplearemos: bigramas y trigramas.

Bigramas

Serán grupos de 2 letras, tomados secuencialmente al recorrer letra por letra cada palabra.

Ejemplos:

Chaymantataq → ch ha ay ym ma an nt ta at ta aq (y después de eso)

Tutapitaqmi → tu ut ta ap pi it aq qm mi (En la noche)

Waqayninta → wa aq qa ay yn ni in nt ta (Llorando)

Al corpus de bigramas lo denominaremos BOBQ.

Trigramas

Serán grupos de 3 letras, tomados secuencialmente al recorrer letra por letra cada palabra.

Ejemplos:

Chaymantataq → cha hay aym yma man ant nta tat ata taq

Tutapitaqmi → tut uta tap api pit ita taq aqm qmi

Waqayninta → waq aqa qay qyn yni nin int nta

Al corpus de trigramas lo designaremos como BOTQ.

Debido a la riqueza que existe en la pronunciación asociada a cada dialecto además de los atributos antes descritos, se admitió considerar la pronunciación, hubiera sido ideal disponer de este recurso para utilizarlo como otro atributo, pero a la fecha de hoy no contamos con una fuente de datos de este tipo. Para trabajos futuros consideraremos necesarios tenerlos en cuenta.

6.4 Descripción de los experimentos

Se combinarán los 5 algoritmos de clasificación seleccionados (*Naive Bayes, Trees j48, Reglas PART, Reglas Jrip, Función SMO*) con las 2 formas de *cross validation* (5 *folds* y 10 *folds*) y con los 4 tipos de atributos (palabras, lemas, bigramas y trigramas). La combinación de estos elementos se realizará en 3 fases secuenciales. Con ello se pretende evitar aquellas combinaciones de las que no se espera mejorar resultados.

Primera fase

La primera fase consta de 3 experimentos, el primero se realizará sobre el corpus original BOW, el segundo sobre el corpus obtenido tras una primera reducción de la dimensionalidad, con sólo palabras cuyos letras pertenezcan al alfabeto quechua y sin

considerar las palabras menos frecuentes en todo el corpus, este corpus se denominará BOWQ y finalmente el corpus obtenido luego de lematizar el BOWQ, que denominaremos BOLQ. Cada uno de los pre-procesos está explicado con más detalle en el capítulo 5 de este trabajo.

Identificador del Experimento	*Atributos*	*Algoritmo*	*Evaluación*
Experimento 1	Corpus original de palabras BOW	Naive Bayes	CV5 y CV10
		Trees j48	
		Reglas PART	
		Reglas Jrip	
		Función SMO	
Experimento 2	Corpus con palabras del alfabeto quechua, sin considerar las palabras de menor frecuencia BOWQ	Naive Bayes	CV5 y CV10
		Trees j48	
		Reglas PART	
		Reglas Jrip	
		Función SMO	
Experimento 3	Corpus con lemas quechua BOLQ	Naive Bayes	CV5 y CV10
		Trees j48	
		Reglas PART	
		Reglas Jrip	
		Función SMO	

Tabla 6.1: Combinación de experimentos para la primera fase.

Segunda fase

El corpus que mejores resultados proporcione de entre los experimentos 1 y 2 de la primera fase será utilizado como base para aplicar el *stop word list* y realizar los siguientes 4 experimentos para ir reduciendo el número de combinaciones posibles. No se tomará el corpus del experimento 3 porque ya se

aplicó la disminución de la dimensionalidad por lematización y no convendría reducirlo aún más.

Se aplicará para esta fase *stop word list* de 200, 150, 100 y 90 palabras, según se detalla en la tabla 6.2.

Identificador del Experimento	*Atributos*	*Algoritmo*	*Evaluación*
Experimento 4	Corpus aplicando el *stop word list* de 200 palabras SL200	Naive Bayes	CV 5 Y CV 10
		Trees j48	
		Reglas PART	
		Reglas Jrip	
		Función SMO	
Experimento 5	Corpus aplicando el *stop word list* de 150 palabras SL150	Naive Bayes	CV5 y CV10
		Trees j48	
		Reglas PART	
		Reglas Jrip	
		Función SMO	
Experimento 6	Corpus aplicando el *stop word list* de 100 palabras SL100	Naive Bayes	CV5 y CV10
		Trees j48	
		Reglas PART	
		Reglas Jrip	
		Función SMO	
Experimento 7	Corpus aplicando el *stop word list* de 90 palabras SL90	Naive Bayes	CV5 y CV10
		Trees j48	
		Reglas PART	
		Reglas Jrip	
		Función SMO	

Tabla 6.2: Combinación de experimentos para la segunda fase.

Tercera fase

Para la tercera fase de experimentos trabajaremos sobre el corpus con mejores resultados de la segunda fase. Se considerarán de este corpus; primero los lemas BOLQ_X (X será la denominación del mejor corpus de la segunda fase), luego bigramas BOBQ y finalmente los trigramas BOTQ.

Identificador del Experimento	*Atributos*	*Algoritmo*	*Evaluación*
Experimento 8	Corpus de Lemas BOLQ_X	Naive Bayes	CV5 y CV10
		Trees j48	
		Reglas PART	
		Reglas Jrip	
		Función SMO	
Experimento 9	Corpus de Bigramas BOBQ	Naive Bayes	CV5 y CV10
		Trees j48	
		Reglas PART	
		Reglas Jrip	
		Función SMO	
Experimento 10	Corpus de Trigramas BOTQ	Naive Bayes	CV5 y CV10
		Trees j48	
		Reglas PART	
		Reglas Jrip	
		Función SMO	

Tabla 6.3: Combinación de experimentos para la tercera fase.

Cuarta fase

Para esta fase se eligen sólo 2 algoritmos, los que hayan proporcionado mejores resultados en las fases anteriores. Estos algoritmos se probarán con todos los atributos combinados entre sí. Dividiremos esta fase en 3 grupos, como sigue:

Grupo 1

Para este grupo sólo consideraremos un experimento, para ello combinaremos todos los 4 atributos: palabras, lemas, bigramas y trigramas. Denominaremos al corpus obtenido BOWLBT.

Identificador del Experimento	*Atributos*	*Algoritmo*	*Evaluación*
Experimento 11	Corpus de Palabras, Lemas, Bigramas y Trigramas BOWLBT	2 mejores algoritmos	CV5 y CV10

Tabla 6.4: Detalle del experimento 11.

Grupo 2

Para cada uno de los experimentos de este grupo, combinaremos los 4 atributos antes mencionados, en combinaciones de a 3. Como en el caso del experimento 11, trabajaremos sobre el SL100. En la siguiente tabla se indican todas estas combinaciones, con el detalle de atributos combinados para cada experimento.

Identificador del Experimento	*Atributos*	*Algoritmo*	*Evaluación*
Experimento 12	Corpus de Palabras, Lemas y Bigramas BOWLB	2 mejores algoritmos	CV5 y CV10
Experimento 13	Corpus de Palabras, Lemas y Trigramas BOWLT		
Experimento 14	Corpus de Palabras, Bigramas y Trigramas BOWBT		
Experimento 15	Corpus de Lemas, Bigramas y Trigramas BOLBT		

Tabla 6.5: Detalle de los experimentos de la tercera fase del grupo 2.

Grupo 3

Por último para el grupo 3 de experimentos, tomaremos los 4 atributos ya mencionados anteriormente, de 2 en 2, como se muestra con detalle en la tabla 6.6.

Identificador del Experimento	*Atributos*	*Algoritmo*	*Evaluación*
Experimento 16	Corpus de Palabras y Lemas BOWL	2 mejores algoritmos	CV5 y CV10
Experimento 17	Corpus de Palabras y Bigramas BOWB		
Experimento 18	Corpus de Palabras y Trigramas BOWT		
Experimento 19	Corpus de Lemas y Bigramas BOLB		
Experimento 20	Corpus de Lemas y Trigramas BOLT		
Experimento 21	Corpus de Bigramas y Trigramas BOBT		

Tabla 6.6: Detalle de experimentos del grupo 3 de la tercera fase.

La idea de esta combinación de corpus más atributos más algoritmos es intentar encontrar el modelo que mejor funcione en la clasificación de dialecto.

Los resultados de los experimentos se presentarán con las medidas de: *Precision, Recall* y *F-measure*, que están definidas en la sección 4.3.

7 RESULTADOS

A continuación se presentan los resultados obtenidos, para los experimentos planteados en la sección 6.

7.1 Primera fase

En la tabla 7.1 se observan los resultados de los 3 primeros experimentos realizados. Los mejores resultados para los experimentos 1 y 2 son los obtenidos al aplicar el algoritmo basado en reglas Jrip con F-Measure de 0.870 y 0.882 respectivamente, mientras que el de peor desempeño en ambos casos resulta ser el Naive Bayes con F-Measure de 0.682 en ambos casos.

Para el experimento 3, el algoritmo con mejor rendimiento es el SVM con un F-Measure de 0.882 y de forma similar que en los experimentos 1 y 2 el algoritmo que obtuvo el peor resultado fue Naive Bayes con un F-Measure de 0.711.

Se observa mejora de los resultados en el experimento 2, en el cual se aplicó técnicas de reducción de la dimensionalidad, básicamente del número de atributos (palabras) al validar las palabras con el alfabeto quechua y al eliminar las palabras que aparecen con menor frecuencia en el corpus original.

	Test →	Clasificador									
		Naive Bayes		Trees (j48)		Reglas (PART)		Reglas (Jrip)		SVM (SMO)	
		CV5	*CV10*	*CV5*	*CV10*	*CV5*	*CV10*	*CV5*	*CV10*	*CV5*	*CV10*
Experimento 1 BOW	**Pr**	0.711	0.699	0.765	0.800	0.788	0.839	0.847	0.870	0.819	0.850
	Re	0.706	0.694	0.765	0.800	0.788	0.835	0.847	0.871	0.812	0.847
	F-M	0.695	0.682	0.765	0.800	0.787	0.833	0.846	**0.870**	0.808	0.845
Experimento 2 BOWQ	**Pr**	0.699	0.699	0.765	0.800	0.788	0.839	0.882	0.835	0.794	0.850
	Re	0.694	0.694	0.765	0.800	0.788	0.835	0.882	0.835	0.788	0.847
	F-M	0.682	0.682	0.765	0.800	0.787	0.833	**0.882**	0.835	0.784	0.845
Experimento 3 BOLQ	**Pr**	0.741	0.719	0.829	0.784	0.777	0.741	0.811	0.788	0.871	0.882
	Re	0.741	0.718	0.824	0.776	0.776	0.741	0.812	0.788	0.871	0.882
	F-M	0.737	0.711	0.820	0.771	0.774	0.737	0.811	0.787	0.870	**0.882**

Tabla 7.1: Resultados de la clasificación primera fase.

Por otro lado los experimentos 2 y 3 obtienen el mismo mejor resultado y con la aplicación del mismo algoritmo, es decir la lematización aplicada en este punto no tuvo mayor impacto.

La figura 7.1 muestra los resultados obtenidos en el primer grupo de experimentos de manera gráfica.

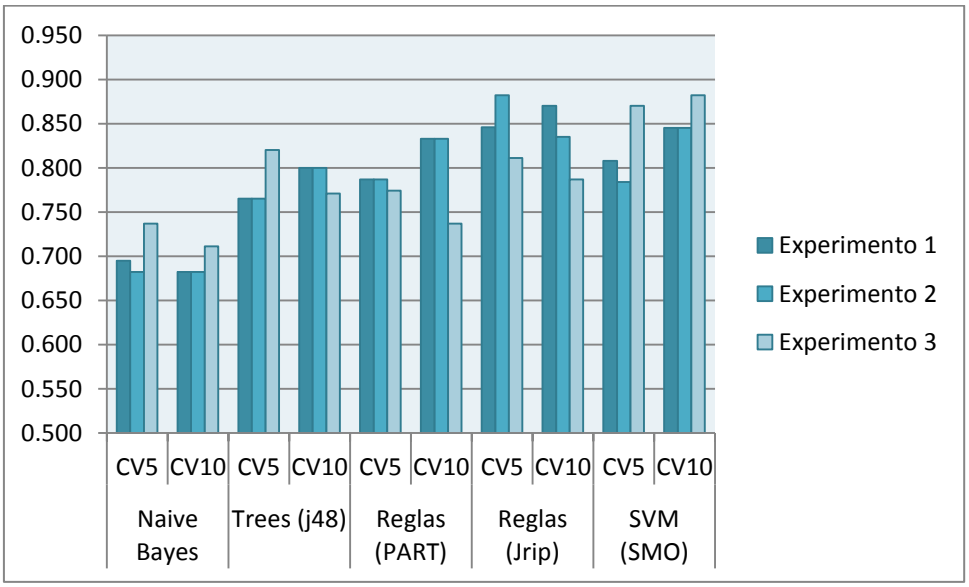

Figura 7.1: Mejora debida a la reducción de la dimensionalidad.

Podemos concluir que para el primer grupo de experimentos de la primera fase, los algoritmos con mejores resultados son el Jrip basado en reglas y el SVM.

7.2 Segunda fase

De la fase anterior, entre los experimentos 1 y 2, el experimento con mejores resultados es el experimento 2, por esta razón para la segunda fase se trabajará sobre el corpus BOWQ puesto que como se menciona en el capítulo 6, para el corpus del experimento 3 ya se aplicó una disminución de la dimensionalidad por lematización y no convendría reducirlo más.

En esta fase se aplican los *stop word list* de 200, 150, 100 y 90 palabras al corpus BOWQ, para los experimentos 4, 5, 6 y 7 respectivamente.

	Test →	Clasificador									
		Naive Bayes		Trees (j48)		Reglas (PART)		Reglas (Jrip)		SVM (SMO)	
		CV5	*CV10*	*CV5*	*CV10*	*CV5*	*CV10*	*CV5*	*CV10*	*CV5*	*CV10*
Experimento 4 SL200	**Pr**	0.715	0.715	0.680	0.754	0.711	0.740	0.692	0.676	0.763	0.784
	Re	0.706	0.706	0.682	0.753	0.706	0.741	0.694	0.671	0.753	0.776
	F-M	0.692	0.692	0.677	0.753	0.695	0.740	0.690	0.653	0.744	**0.771**
Experimento 5 SL150	**Pr**	0.715	0.715	0.710	0.764	0.767	0.769	0.753	0.799	0.763	0.784
	Re	0.706	0.706	0.706	0.765	0.765	0.765	0.753	0.788	0.753	0.776
	F-M	0.692	0.692	0.707	0.764	0.765	0.759	0.750	**0.782**	0.744	0.771
Experimento 6 SL100	**Pr**	0.715	0.715	0.776	0.788	0.788	0.825	0.895	0.906	0.773	0.809
	Re	0.706	0.706	0.776	0.788	0.788	0.824	0.894	0.906	0.765	0.800
	F-M	0.692	0.692	0.776	0.787	0.787	0.822	0.893	**0.906**	0.757	0.795
Experimento 7 SL90	**Pr**	0.699	0.699	0.765	0.800	0.788	0.839	0.882	0.835	0.794	0.850
	Re	0.694	0.694	0.765	0.800	0.788	0.835	0.882	0.835	0.788	0.847
	F-M	0.682	0.682	0.765	0.800	0.787	0.833	**0.882**	0.835	0.784	0.845

Tabla 7.2: Resultados de la clasificación de la segunda fase.

El mejor resultado para el experimento 4 es el F-Measure de 0.771 obtenido con el algoritmo SVM y el peor de 0.653 obtenido con el algoritmo Jrip.

Para el experimento 5 el algoritmo con mejor rendimiento resulta ser el Jrip con F-Measure de 0.782 y el de peor desempeño el Naive Bayes con 0.692 de F-Measure.

En el caso del experimento 6 el mejor resultado es el F-Measure de 0.906 obtenido con el algoritmo Jrip y el peor F-Measure es de 0.692 obtenido con el algoritmo Naive Bayes.

El mejor resultado para el experimento 7 es el F-Measure de 0.882 logrado con el algoritmo Jrip y el peor F-Measure es 0.682 obtenido con Naive Bayes.

Podemos señalar que los valores de *Precision, Recall* y *F-Measure* son más altos cuantas menos palabras posee el *stop word list*, hasta un punto de corte de 100 palabras obteniendo un *F-Measure* de 0.906; puesto que ya con el de 90 palabras se logra un *F-Measure* de 0.882.

Al igual que ocurre en los experimentos anteriores, observamos que los mejores algoritmos de clasificación en esta fase son el Jrip y el SVM. En general bastante equilibrados *Pr* y *Re*.

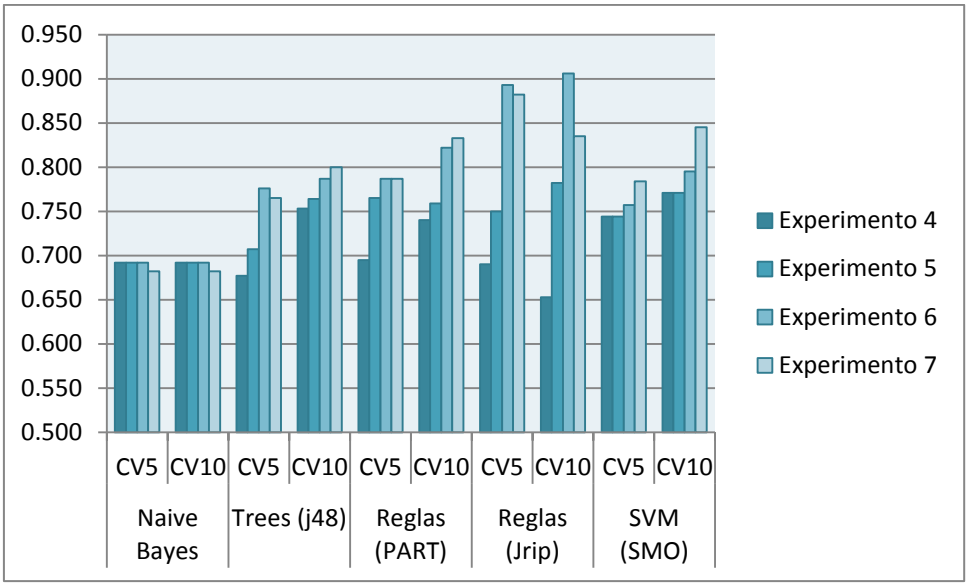

Figura 7.2: Influencia en los resultados *F-Measure* del tamaño de la lista de palabras del *stop word list*.

En la figura 7.2 se observa mejores resultados del *F-Measure* para el experimento 6 (SL100) con un incremento de más de 10 puntos sobre el resto.

Podemos concluir que la aplicación de un *stop word list* de 100 palabras tuvo un impacto positivo en los resultados.

7.3 Tercera fase

Para la tercera fase de experimentación trabajaremos sobre el corpus SL100, por ser el de mejores resultados hasta este punto.

Para los experimentos 8, 9 y 10 de esta fase emplearemos corpus de lemas, bigramas y trigramas respectivamente, esto con el fin de comparar el rendimiento de cada uno de ellos.

	Test →	Clasificador									
		Naive Bayes		Trees (j48)		Reglas (PART)		Reglas (Jrip)		SVM (SMO)	
		CV5	*CV10*	*CV5*	*CV10*	*CV5*	*CV10*	*CV5*	*CV10*	*CV5*	*CV10*
Experimento 8 BOLQ	**Pr**	0.741	0.719	0.692	0.770	0.756	0.780	0.729	0.713	0.847	0.859
	Re	0.741	0.718	0.694	0.765	0.753	0.776	0.729	0.706	0.847	0.859
	F-M	0.737	0.711	0.691	0.766	0.754	0.777	0.729	0.707	0.847	**0.859**
Experimento 9 BOBQ	**Pr**	0.765	0.765	0.835	0.882	0.801	0.860	0.913	0.884	0.824	0.871
	Re	0.753	0.753	0.835	0.882	0.800	0.859	0.906	0.882	0.824	0.871
	F-M	0.754	0.754	0.835	0.882	0.800	0.859	**0.904**	0.881	0.822	0.870
Experimento 10 BOTQ	**Pr**	0.765	0.765	0.730	0.741	0.800	0.764	0.705	0.756	0.898	0.888
	Re	0.765	0.765	0.729	0.741	0.800	0.765	0.706	0.753	0.894	0.882
	F-M	0.765	0.765	0.730	0.737	0.800	0.764	0.705	0.754	**0.893**	0.881

Tabla 7.3: Resultados de la clasificación segunda fase grupo 1.

Para el experimento 8 de corpus de lemas, el mejor *F-Measure* obtenido es 0.859 con el algoritmo SVM y el peor *F-Measure* es 0.691 con el j48. En el caso del corpus de bigramas en el experimento 9, el mejor desempeño es del algoritmo Jrip con un *F-Measure* de 0.904 y el peor es el *F-Measure* de 0.754 con el algoritmo Naive Bayes. Para el caso del corpus de trigramas del experimento 10 el algoritmo con mejor desempeño es SVM con *F-Measure* de 0.893 y el de peor rendimiento es Jrip con *F-Measure* de 0.705.

Los algoritmos con mejor desempeño en esta fase resultan ser Jrip y SMO. Por otro lado, ninguno de los experimentos de esta fase supera los resultados (0.906) obtenidos en la fase anterior con el corpus SL100 y algoritmo Jrip.

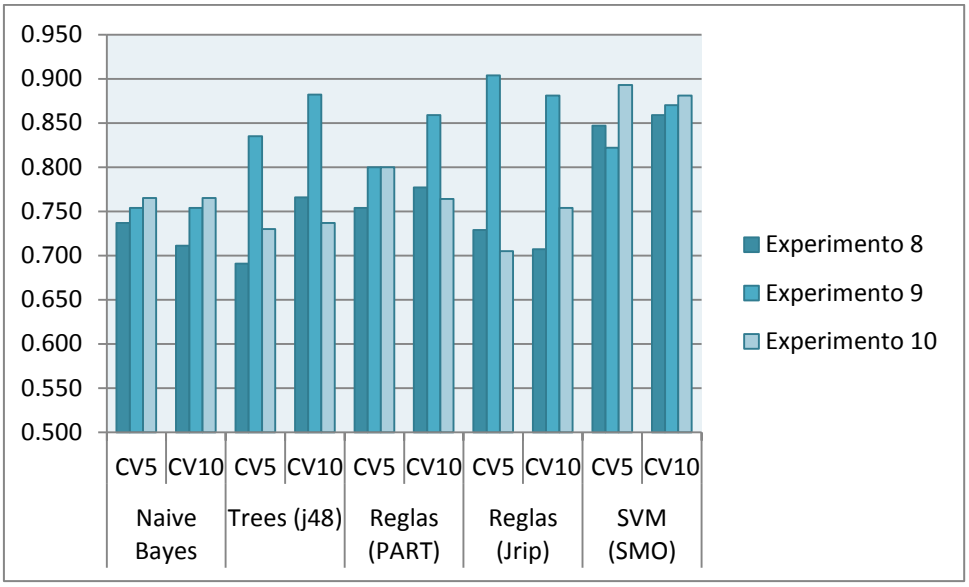

Figura 7.3: Comparación de los resultados *F-Measure* entre lemas, bigramas y trigramas.

Se puede observar que no ha habido mejoría en los resultados, sino todo lo contrario con la aplicación de lemas en el experimento 8, bigramas en el experimento 9 y trigramas en el experimento 10. Los algoritmos con mejores resultados siguen siendo Jrip y SMO.

Para pasar a la cuarta fase se eligen sólo 2 algoritmos, los que lograron mejores resultados en las 3 fases 1, 2 y 3, para ello se presenta a continuación una tabla con el resumen de todos los experimentos realizados hasta este punto, sólo con los *F-Measure* obtenidos.

	Test →	Clasificador									
		Naive Bayes		Trees (j48)		Reglas (PART)		Reglas (Jrip)		SVM (SMO)	
		CV5	*CV10*	*CV5*	*CV10*	*CV5*	*CV10*	*CV5*	*CV10*	*CV5*	*CV10*
Experimento 1 *BOW*	**F-M**	0.695	0.682	0.765	0.800	0.787	0.833	0.846	**0.870**	0.808	0.845
Experimento 2 *BOWQ*	**F-M**	0.682	0.682	0.765	0.800	0.787	0.833	**0.882**	0.835	0.784	0.845
Experimento 3 *BOLQ*	**F-M**	0.737	0.711	0.820	0.771	0.774	0.737	0.811	0.787	0.870	**0.882**
Experimento 4 *SL200*	**F-M**	0.692	0.692	0.677	0.753	0.695	0.740	0.690	0.653	0.744	**0.771**
Experimento 5 *SL150*	**F-M**	0.692	0.692	0.707	0.764	0.765	0.759	0.750	**0.782**	0.744	0.771
Experimento 6 *SL100*	**F-M**	0.692	0.692	0.776	0.787	0.787	0.822	0.893	**0.906**	0.757	0.795
Experimento 7 *SL90*	**F-M**	0.682	0.682	0.765	0.800	0.787	0.833	**0.882**	0.835	0.784	0.845
Experimento 8 *BOLQ2*	**F-M**	0.737	0.711	0.691	0.766	0.754	0.777	0.729	0.707	0.847	**0.859**
Experimento 9 *BOBQ*	**F-M**	0.754	0.754	0.835	0.882	0.800	0.859	**0.904**	0.881	0.822	0.870
Experimento 10 *BOTQ*	**F-M**	0.765	0.765	0.730	0.737	0.800	0.764	0.705	0.754	**0.893**	0.881

Tabla 7.4: Resumen de los mejores resultados de la primera, segunda y tercera fase.

Se puede apreciar claramente en la tabla 7.4 el mejor desempeño de los algoritmos Jrip y SMO para la mayoría de los experimentos realizados en la primera, segunda y tercera fase, a diferencia del algoritmo basado en arboles J48 y el algoritmo basado en reglas PART, que solo presentan desempeños óptimos en algunos casos;

Por otro lado cabe señalar que los experimentos con los valores más altos de *F-Measure*, son el experimento 6 en el que se aplicó un *stop word list* de 100 palabras, obteniendo un *F-Measure* de 0.906 y el experimento 9 en el cuál se aplicó bigramas sobre el corpus del experimento 6 obteniendo un *F-Measure* de 0.904 que no logra superar al anterior, pero que es muy similar.

Por lo tanto para la tercera fase sólo se emplearán Jrip y SMO para evitar todas las combinaciones.

7.4 Cuarta fase

Grupo 1

A continuación en la tabla 7.5 se muestran los resultados obtenidos tras realizar la combinación de los 4 atributos; palabras, lemas, bigramas y trigramas. Se emplearon los algoritmos Jrip y SMO.

	Test →	Clasificador			
		Reglas (Jrip)		SVM (SMO)	
		CV5	*CV10*	*CV5*	*CV10*
Experimento 11	**Pr**	0.859	0.859	0.836	0.839
	Re	0.859	0.859	0.835	0.835
	F-M	**0.858**	**0.858**	0.834	0.833

Tabla 7.5: Resultados de la clasificación en la cuarta fase grupo 1.

De los resultados conseguidos se puede señalar que combinando los cuatro atributos no se logra superar a los mejores resultados obtenidos en las fases anteriores. El algoritmo con mejor eficacia fue el algoritmo basado en reglas Jrip, con iguales valores de evaluación para el *cross validation* de 5 y 10 *folds*. Aun así entre estos no se observa mejoras significativas puesto que no superan los 5 puntos.

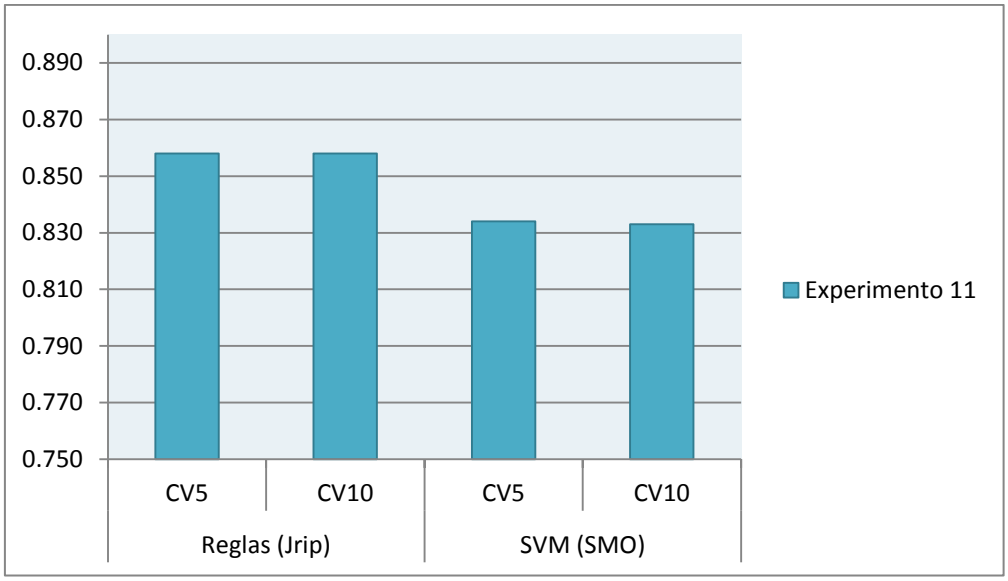

Figura 7.4: Resultados para la combinación de todos los atributos.

La figura 7.4 ayuda a visualizar de mejor forma los resultados obtenidos al combinar los 4 atributos.

Grupo 2

En la tabla 7.6 se muestran los resultados de la clasificación utilizando los diferentes corpus obtenidos tras combinar los atributos: palabras, lemas, bigramas y trigramas en combinaciones de a 3.

	Test →	Clasificador			
		Reglas (Jrip)		SVM (SMO)	
		CV5	*CV10*	*CV5*	*CV10*
Experimento 12	**Pr**	0.837	0.906	0.825	0.860
	Re	0.835	0.906	0.824	0.859
	F-M	0.836	**0.906**	0.822	0.858
Experimento 13	**Pr**	0.743	0.767	0.906	0.918
	Re	0.741	0.765	0.906	0.918
	F-M	0.742	0.765	0.906	**0.918**
Experimento 14	**Pr**	0.765	0.859	0.850	0.850
	Re	0.765	0.859	0.847	0.847
	F-M	0.765	**0.858**	0.845	0.845
Experimento 15	**Pr**	0.777	0.814	0.847	0.864
	Re	0.776	0.812	0.847	0.859
	F-M	0.777	0.812	0.846	**0.857**

Tabla 7.6: Resultados de la clasificación en la cuarta fase grupo 2.

El algoritmo con mejor rendimiento para los experimentos 12 y 14 es el Jrip con medidas de F-Measure de 0.906 y 0.858 respectivamente. Mientras que para los experimentos 13 y 15 el mejor algoritmo resulta ser SVM con medidas de F-Measure de 0.918 y 0.857 respectivamente.

Se aprecia que combinando los 4 atributos de a 3, si se ha logrado un impacto positivo en los resultados. El algoritmo clasificador con mejor desempeño es el SMO con la opción de evaluación Cross Validation de 10 folds.

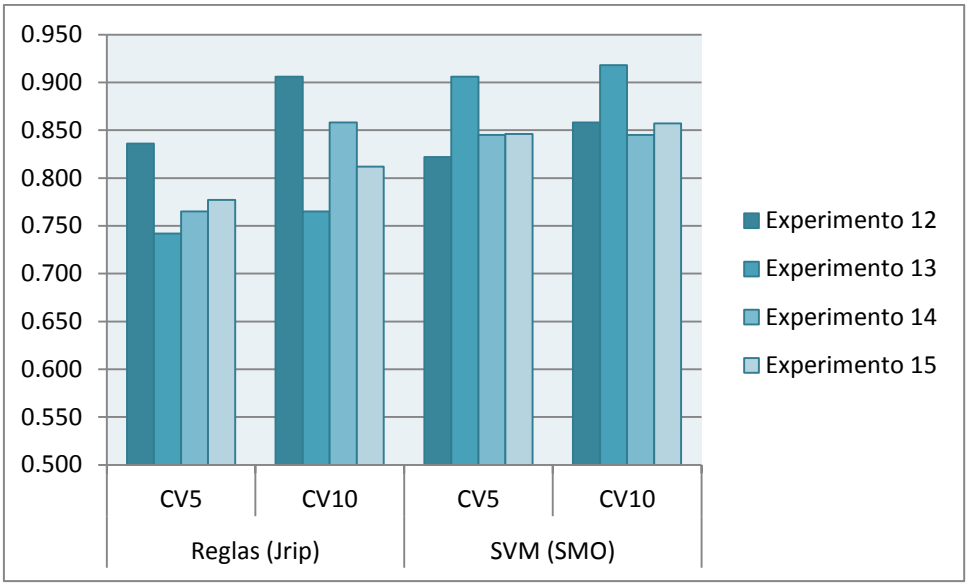

Figura 7.5: Resultados de las combinaciones de a 3 de los atributos.

En la figura 7.5 se observa la mejoría de los resultados para las combinaciones de atributos del experimento 12 (palabras, lemas y bigramas) y del experimento 13 (palabras, lemas y trigramas). En ambos casos los resultados son más que satisfactorios.

Grupo 3

A continuación en la tabla 7.7 se muestran los resultados de los experimentos realizados, combinando los 4 atributos de a 2.

	Test →	Clasificador			
		Reglas (Jrip)		SVM (SMO)	
		CV5	*CV10*	*CV5*	*CV10*
Experimento 16	**Pr**	0.723	0.859	0.835	0.871
	Re	0.718	0.859	0.835	0.871
	F-M	0.719	0.859	0.835	**0.870**
Experimento 17	**Pr**	0.918	0.919	0.884	0.906
	Re	0.918	0.918	0.882	0.906
	F-M	**0.918**	0.917	0.881	0.906
Experimento 18	**Pr**	0.746	0.733	0.871	0.884
	Re	0.741	0.729	0.871	0.882
	F-M	0.742	0.730	0.870	**0.881**
Experimento 19	**Pr**	0.874	0.870	0.825	0.847
	Re	0.871	0.871	0.824	0.847
	F-M	0.869	**0.870**	0.822	0.846
Experimento 20	**Pr**	0.827	0.788	0.895	0.895
	Re	0.824	0.788	0.894	0.894
	F-M	0.824	0.788	**0.893**	**0.893**
Experimento 21	**Pr**	0.732	0.884	0.850	0.864
	Re	0.718	0.882	0.847	0.859
	F-M	0.719	**0.883**	0.845	0.857

Tabla 7.7: Resultados de la clasificación en la cuarta fase grupo 3.

El algoritmo con mejor desempeño en los experimentos 16, 18 y 20 es el SVM con medidas de F-Measure de 0.870, 0.881 y 0.893 respectivamente.

Para los experimentos 17, 19 y 20 el algoritmo con mejor rendimiento es el Jrip con medidas de F-Measure iguales a 0.918, 0.870 y 0.883 correspondientemente.

Se distinguen mejores resultados para el experimento 17 con el algoritmo basado en reglas Jrip para un *cross validation* de 5 *folds* logrando un *F-Measure* de 9.18 seguido de 9.17 para el mismo algoritmo con un *cross validation* de 10 *folds*.

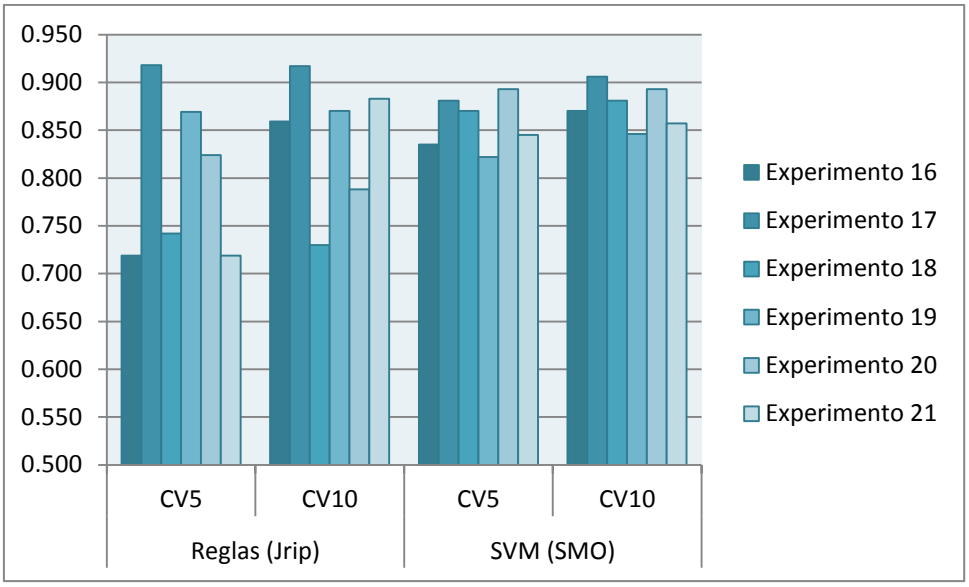

Figura 7.7: Resultados de las combinaciones de a 2 de los atributos.

Los resultados demuestran que combinando palabras con bigramas como se realizó en el experimento 17, se logran mejores resultados que con el resto de combinaciones. El algoritmo con mejor rendimiento en este grupo de experimentos resulto ser el Jrip para una opción de evaluación de 5 *folds* del *cross validation*, logrando un *F-Measure* de 0.918.

Después de culminados todos los experimentos podemos señalar que los que mejores resultados se obtuvieron con el experimento 13 combinando palabras lemas y trigramas con el algoritmo SMO con *cross validation* de 10 *folds* y el experimento 17 combinando los atributos palabras con bigramas con el algoritmo basado en reglas Jrip con *cross validation* de 5 *folds*, ambos con un *F-Measure* de 0.918.

8 CONCLUSIONES Y TRABAJO FUTURO

8.1 Conclusiones

El objetivo principal que se ha planteado inicialmente, que fue el de construir un clasificador automático de textos para el quechua cusqueño ha sido conseguido.

La tarea de conseguir un corpus amplio de textos escritos en quechua, es una tarea en la que se sigue trabajando hasta la fecha. Este trabajo de recopilación es muy importante para abordar las tareas del procesamiento de lenguaje natural quechua, alimentar la base de datos léxica quechua, mejorar el corrector ortográfico de textos, etc.

Los mejores clasificadores que se han conseguido, fueron los que emplearon como atributos palabras combinadas con lemas y trigramas, empleando el algoritmo SMO, y usando la opción de entrenamiento del *cross validation* de 10 *folds* con un *F-Measure* de 0.918. Al igual que el clasificador que empleó la combinación de palabras con trigramas con el uso del algoritmo Jrip y con la opción de entrenamiento del *cross validation* de 5 *folds* logrando también un *F-Measure* de 0.918.

Las técnicas empleadas para la reducción de la dimensionalidad del corpus, mejoraron los resultados. Algunas de estas técnicas tuvieron un impacto directo en los resultados; como es el caso de los pre-procesos para la limpieza del texto (caracteres extraños, palabras que contenían letras no pertenecientes al alfabeto quechua, palabras con menor frecuencia de aparición), y la aplicación de los *stop word list* hasta un punto de corte de 100 palabras. Por otro lado, técnicas como la lematización y el uso de n-gramas (bigramas y trigramas) lograron mejorar los resultados, aunque no directamente, sino a través de la combinación de éstas.

8.2 Trabajos futuros

Se sugiere para trabajos futuros la combinación de los algoritmos de clasificación, debido a que es una práctica frecuentemente empleada para la construcción de clasificadores y que logra mejorar los resultados en algunos casos. En este trabajo se optó por la combinación de atributos en lugar de clasificadores.

También se sugiere, para trabajos futuros, el uso de la pronunciación como atributo, debido a la riqueza de características que posee asociada a cada dialecto quechua.

Se sugiere como trabajo futuro la construcción de un clasificador de textos para el dialecto sureño del quechua.

El problema de la estandarización del quechua en el Perú, es un problema que se debe afrontar para la realización de trabajos de procesamiento de lenguaje natural en el futuro. Como se mencionó en el capítulo 2 de este trabajo existe una disputa, debido al problema del tri y penta vocalismo en el quechua. Existen instituciones sólidas que se manifiestan a favor del penta vocalismo, como es el caso de La Real Academia de la Lengua Quechua. Por el otro lado el lingüista Rodolfo Cerrón Palomino, destacado quechuista peruano y miembro de la Academia Peruana de la Lengua Española ha establecido el estándar del "Quechua Sureño Unificado", único estándar existente en el Perú a la fecha.

Sería conveniente que el corpus este en formato XML para facilitar su acceso en posteriores trabajos.

Queda también pendiente el desarrollo de una aplicación web de clasificación automática de textos en quechua cusqueño.

9 BIBLIOGRAFÍA

[1] Academia Mayor de la Lengua Quechua. Qoriwata. *Municipalidad Provincial del Cusco,* 2003.

[2] A. Rios, Applying Finite-State Techniques to a Native American Language: Quechua*. Herbstsemester,* 2010.

[3] A. Cusihuaman. Gramática Quechua Cuzco Collao. *Centro de Estudios Regionales Andinos "Bartolomé de las Casas"*, 2001.

[4] Diccionario de la Real Academia de la Lengua Quechua, *Gobierno Regional Cusco, Perú,* 2006.

[5] D. Gonzáles de Holguín. Arte y Diccionario Qquechua-Español. *Calle de la Rifa, Lima,* 1901.

[6] E. Riloff. Little words can make a big difference for text classification. In Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pag. 130--136, 1997.

[7] F. Sebastiani. Machine Learning in Automated Document Categorization. *The 18th International Conference on Computational Linguistics. Tutorials, Nancy, Francia,* 2000.

[8] http://www.cs.waikato.ac.nz/ml/weka/

[9] I. Dagan, Y. Karov y D. Roth. Mistake- Driven Learning in Text Categorization. *EMNLP '97, 2nd Conference on Empirical Methods in Natural Language Processing,* 1997.

[10] J. Ayala, M. Ballena Dávila, J. Chávez, M. Chávez, D. Coombs, A. Koop, D. Koop, G. López de Hoyos, C. Parker. Pueblos del Perú. *Instituto Lingüístico de Verano. Javier Prado Oeste 200, Magdalena. Casilla 2492, Lima 100, Perú,* 2006.

[11] J. A. Lira. Diccionario Kkechua – Español. *Universidad Nacional de Tucumán, Argentina,* 1944.

[12] J. Calvo Pérez. Pragmática y Gramática del Quechua Cuzqueño. *Pampa de la alianza 465, Perú,*1993.

[13] K. Hornik, P. Mair y J. Rauch. The textcat Package for n-Gram Based Text Categorization in R. *Journal of Statistical Software Volume 52, Issue 6,* 2013.

[14] O. Arregi e I. Fernández. Clasificación de documentos escritos en euskara: impacto de la Lematización.

[15] P. Náther. N-gram based Text Categorization. *Comenius University,* 2005.

[16] P.P.T.M. van Mun. Text Classification in Information Retrieval using Winnow. url: citeseer.nj.nec.com/133034.html, 1999.

[17] R. Krovetz. Viewing morphology as an inference process. In Proceedings of ACMSIGIR93, pag. 191--203. 1993.

[18] S. Chakrabarti. Mining the Web: discovering knowledge from hypertext data. *Morgan Kaufmann, San Francisco, CA, cop.* 2003.

[19] S. M. Weiss, N. Indurkhya, T. Zhang, F. J. Damerau. *Text Mining Predictive Methods for Analyzing Unstructured Information. United States of America,* 2005.

[20] W. B. Cavnar y J. M. Trenkle. N-gram-based text categorization. *Proceedings of SDAIR- 94, 3rd Annual Symposium on Document Analysis and Information Retrieval*, 1994.

[21] Z. Figueroa Cusi, D. Tunque Choque. Manual para el aprendizaje del idioma quechua. *Cusco, Perú,* 2009.

ANEXOS

ANEXO 1

N°	Nombre del documento	N° palabras original	Dialecto	Origen	Tipo	Fecha
1	Unidos	137	Ancash	CDM	Cuento	No se conoce
2	José maríawan casarakusqanmanta	509	Apurimac	UNIA	Cuento	2010
3	Wayna siipaskuna sumaq p'achakusqankumantta wiillllakuy	307	Apurimac	UNIA	Cuento	2010
4	Ukumariq kawsayninmanta	361	Apurimac	UNIA	Cuento	2010
5	Don tellessfforomantta ssumaq wiillllakuy	359	Apurimac	UNIA	Cuento	2010
6	Atoqmanta huk'uchamantawan	235	Apurimac	UNIA	Cuento	2011
7	Warmachamanta sumaq willakuy	287	Apurimac	UNIA	Cuento	2010
8	Apurimakpa runasimin i	1,504	Apurimac	ALQA	Escolares	2006
9	Apurimakpa runasimin ii	113	Apurimac	ALQA	Escolares	2006
10	Quechuapa onomatopeyankuna	1,309	Apurimac	AIDIA	Cuento	2010
11	Q'enko	228	Apurimac	UNIA	Cuento	2010
12	Sumaq rimaykuna	656	Apurimac	UNIA	Cuento	2010
13	Takakuq ch'aku kapimanta	391	Apurimac	UNIA	Cuento	2010
14	Quechua rimayninchista allinta wiñarichinapaq	967	Apurimac	UNIA	Cuento	2010
15	Llaqtakunatapi umallikunapaq	434	Apurimac	UNIA	Cuento	2010
16	Ñawpa willanakuy	440	Apurimac	CHIRAPAQ	Cuento	2008
17	Kicharisqa ñawiywanmi ñuqa uyarini	94	Chanka	MINEDU	Cuento	2008
18	Pawqar llikllachay	121	Chanka	MINEDU	Cuento	2008
19	Añañaw	42	Chanka	MINEDU	Cuento	2008
20	Hanaq pachaman willakuy qispiq atuqmanta	340	Chanka	MINEDU	Cuento	2008
21	Muki	384	Chanka	MINEDU	Cuento	2008
22	Apus	182	Cusco	CDM	Cuento	No se conoce
23	Atoqmanta wallpamantawan	157	Cusco	LENGAMER	Cuento	2006
24	Atuqcha rakiynin 1	697	Cusco	BNP	Cuentos	2000

25	Atuqcha rakiynin 2	854	Cusco	BNP	Cuentos	2000
26	Atuqcha rakiynin 3	822	Cusco	BNP	Cuentos	2000
27	Ayllunchispi kasqanwan sañukunata rurasun	905	Cusco	CRAM II	Escolares	2000
28	Biblia	453,632	Cusco	RUNASIMI	Biblia	1995
29	Constitucion	12,880	Cusco	CRP	Constitución	1993
30	Huk sisichaq kawsayninmata willakuy	432	Cusco	LENGAMER	Cuento	2006
31	Willakuykuna	2,187	Cusco	ATEK		2007
32	Oveja michiq joseycha	1,559	Cusco	LENGAMER	Cuento	2005
33	Fabula	213	Cusco	CDM	Cuento	No se conoce
34	Gripe aviar nisqamanta kolka aylluq ayqekuynin.	1,148	Cusco	SIL	Escolares	2005
35	Inka taytanchiskunaq	1,645	Cusco	CRAM II	Escolares	2002
36	Kawsayninchis qallariynin	921	Cusco	CRAM II	Escolares	2002
37	Yachay q'ipikuna	3,136	Cusco	FEL	Escolares	2009
38	Llaqtanchispa sunqun	2,077	Cusco	CRAM II	Escolares	2002
39	Makinchiswan Ruranchis	1,639	Cusco	CRAM II	Escolares	2002
40	Yachaywasi ukupi rimayninchiktawan yachayninchiktawan chaninchasunchik	2,956	Cusco	FEL	Escolares	2009
41	Qosqo-puno qheswa qelqayta ñawinchayta Yachaqana	270	Cusco	Pr. Ricardo Cahuana Q.	Tutorial	2007
42	Ocasiones para Aprender	401	Cusco	UNICEF	Tutorial	No se conoce
43	Munaychata pukllasun	1,755	Cusco	CRAM II	Escolares	2000
44	Pacha mamanchispa qhapaq kaynin	1,462	Cusco	CRAM II	Escolares	2000
45	Pisqantin yuyaykunaq atipaynin	294	Cusco	CRAM II	Escolares	2000
46	Poemas en quechua de cusco collao	430	Cusco	Juan de la Cruz Huanta	Poemas	No se conoce
47	Pukllaspa hamut'anchis	488	Cusco	CRAM II	Escolares	2000
48	Sistema de comunicación quechua	125	Cusco	SIL	Tutorial	2006
49	Takiyninchiskunata t'ikarichiq instrumintukuna	951	Cusco	CRAM II	Escolares	2002

50	Teqse	15,215	Cusco	Universidad Ricardo Palma	Académico	2010
51	T'ika	1,966	Cusco	ISPPTA	Escolares	2004
52	Rimayninchispa rimariynin	68	Cusco	LENGAMER	Tutorial	2007
53	Wayk'uq irqikuna	479	Cusco	CRAM II	Escolares	2002
54	Yachay q'ipikuna	401	Cusco	MIB	Escolares	2010
55	Yuyayninchispi paqarichisqa pukllaykuna	1,471	Cusco	CRAM II	Escolares	2002
56	Takisun tususun	1,323	No se conoce	MINEDU	Tutorial	2008
57	Kichasha ñawiwan, nuqa rikaa	47	Huanuco	MINEDU	Cuento	2008
58	Unay wata ñawpata	178	Huanuco	MINEDU	Cuento	2008
59	Tanta wawapa hatun punchawnin	409	Huanuco	MINEDU	Cuento	2008
60	Huk atuqsi munasqa killapa warma yanan kayta	197	Huanuco	MINEDU	Cuento	2008
61	Intimpa,intipa mallkin	394	Huanuco	MINEDU	Cuento	2008
62	Awila mikayla	235	Huanuco	MINEDU	Cuento	2008
63	Nunash, la bella durmiente	325	Huanuco	MINEDU	Cuento	2008
64	¿Imana kanka?	23	Lambayeque	INKAWASI	Tutorial	2007
65	Atun rimananchikta riqsishun	309	Lambayeque	INKAWASI	Tutorial	2007
66	Inapmanta	38	Lambayeque	INKAWASI	Tutorial	2007
67	Killa	25	Lambayeque	INKAWASI	Tutorial	2007
68	Kunya	24	Lambayeque	INKAWASI	Tutorial	2007
69	Suyukuna tantakasha	254	Lambayeque	INKAWASI	Tutorial	2007
70	Paramuta qatiċhanapaq	38	Lambayeque	INKAWASI	Tutorial	2007
71	Anmalkuna mikun	48	Lambayeque	INKAWASI	Tutorial	2007
72	Tayta quijote sanchuwan purishan	988	Lambayeque	INKAWASI	Tutorial	2007
73	T'ika 1	918	Puno	CARE PERÚ	Escolares	2007
74	T'ika 2	3,525	Puno	CARE PERÚ	Escolares	2007
75	T'ika 3	3,712	Puno	CARE PERÚ	Escolares	2007
76	T'ika 4	402	Puno	CARE PERÚ	Escolares	2007

77	T'ika 5	3,943	Puno	CARE PERÚ	Escolares	2007
78	T'ika 6	7,601	Puno	CARE PERÚ	Escolares	2007
79	Cuentos	20,118	Cusco	CDM	Narración	2010
80	Gregorio	23,939	Cusco	CDM	Narración	2010
81	Urubamba	25,657	Cusco	CDM	Narración	2010
82	Uchum tantiapacächiman	257	Wanca	UNIA	Narración	2010
83	Wancacunap casaracuynin	387	Wanca	UNIA	Narración	2010
84	Allinman caminaycuy	234	Wanca	UNIA	Narración	2010
85	Mishqui munacuy asutiwanmi licun	329	Wanca	UNIA	Narración	2010

UNIA Trabajo realizado en la Universidad Nacional Intercultural de la Amazonía

AIDIA Asociación Interdenominacional para el Desarrollo Integral de Apurímac

CRAM II Instituto Superior "La Salle" -PROYECTO CRAM II

CDM http://conversationsdumonde.blogspot.com/

ISPPTA Instituto Superior Pedagógico Público Túpac Amaru.

ALQA Academia de la Lengua Quechua filial Apurímac

INKAWASI INKAWASI – LAMBAYEQUE

SIL SIL International y Universidad Ricardo Palma

CHIRAPAQ Chirapaq, Centro de Culturas Indígenas el Perú

MINEDU Ministerio de Educacion

BNP BibliotecaNacional del Peru

LENGAMER Lenguas de las Americas, www.lengamer.org