**Sense Embeddings e Sense Identification Task**

* Semantica lessicale
* Risorse lessicali
* Processo di sviluppo di una risorsa lessicale (LESSLEX)
* Processo dietro la formalizzazione di un nuovo problema nella ricerca (Sense embeddings)

**Semantica lessicale:**

Che cosa è? E’ il ramo dell’elaborazione del linguaggio naturale che si occupa della rappresentazione e dell’utilizzo dei sensi delle parole, l’obiettivo di base è stabilire quali sono i sensi degli elementi lessicali presenti in testi, frasi, articoli... Non è semplice. A cosa serve? Pensiamo ad assistenti vocali, motori di ricerca... nella maggior parte dei casi siamo noi umani che ci adattiamo al linguaggio che può capire la macchina. Con un assistente vocale potremmo ammettere una forma di traduzione da linguaggio naturale e rappresentazione logica, ci risparmieremo del lavoro. Le macchine non sono brave a capire i sensi delle parole, a volte ci fanno perdere tempo. Il linguaggio naturale è ambiguo, non è semplice capirlo per una macchina.

**Proprietà del linguaggio naturale:**

* Ambiguità
* Economicità (L’ambiguità ci permette di comunicare molti sensi con poche parole senza aver bisogno di un contesto e senza ripartire sempre da capo)
* Polisemia dei termini. Termini che possono avere più di un senso. (termine bank) Per gli esseri umani è una proprietà perchè permette di ridurre la lingua, basta assegnare un senso in più usando la stessa forma ortografica. La macchina deve aver bisogno di un contesto per disambiguare i significati di parole polisemiche.
* Ambiguità e polisemia sono proprietà strettamente legate.

Come si rappresentano i sensi? La semantica lessicale lo fa con le risorse lessicali! Che cosa sono le risorse lessicali? Sono database, dizionari, formati da unità lessicali (termini, sensi delle parole, multiword expression) associate ad una definizione. La definizione può essere banale come quella data da un dizionario. Le risorse sono codificate in una certa lingua. Le unità lessicali sono tradotte in una lingua o in più lingue. Le risorse lessicali sono il mezzo con cui risolviamo il problema dell’ambiguità e della polisemia.

Esistono 2 macro famiglie di risorse che rappresentano i sensi delle parole:

* **Reti semantiche** (**wordnet**) La conoscenza è rappresentata sottoforma di grafo. I nodi del grafo sono i sensi delle parole. I nodi sono connessi da relazioni semantiche e lessicali (ISA, Part of) I nodi del grafo sono rappresentati da insiemi di termini (synset: synonym set) fra loro sinonimi in un senso. Le unità lessicali sono associate ad una definizione che serve per individuare il senso associato.
* **Risorse distribuzionali:** Prendono il nome dalla teoria distribuzionale che ci dice che termini che occorrono in contesti simili (co-occorrono) tendono ad avere un significato simile. Le due parole (table e desk) probabilmente hanno lo stesso significato. Esempi:
  + I ate my lunch at the table.
  + I ate my lunch at the desk

I sensi delle parole sono rappresentati come vettori numerici (word embeddings) e sono associati alle parole. Vengono costruiti in base alle co-occorrenze. Per costruire il vettore embeddings per la parola “chair” prendo un testo, guardo tutte le frasi in cui occorre il termine “chair” e prendo tutte queste frasi, (indipendentemente dal senso che ha chair in ognuna), le metto insieme, conto le parole con cui chair occorre e formo una sua rappresentazione vettoriale.

**Vantaggi e svantaggi:**

* Reti semantiche:
  + **Specificità lessicale alta**: sono in grado di distinguere facilmente i vari sensi dei termini
  + Grazie al fatto che ci riferiamo con synset non associamo una lingua particolare, siamo in grado di parlare di quel senso in **qualsiasi lingua.**
  + **Le reti semantiche non sono semplici da usare.** Normalmente queste risorse vengono usate per confrontare sensi di termini. Quindi calcolare un punteggio numerico che esprima quanto 2 sensi sono simili. (matita, penna) Confrontare 2 sensi fra loro in un grafo vuol dire in alcuni casi calcolare il percorso minimo tra due nodi che non è sempre accurato o facile e sicuramente è influenzato dalle scelte di costruzione di quel grafo.
* Risorse distribuzionali:
  + Si possono **confrontare facilmente** 2 sensi fra loro, usando similarità del coseno o distanza di euclide. Viene restituito un numero che misura la similarità fra i 2 vettori.
  + Il calcolo della **similarità vettoriale è molto più veloce** della similarità sul grafo delle reti semantiche.
  + **Perdiamo specificità lessicale**: tutti i sensi di una parola sono condensati in un’unica rappresentazione vettoriale.

**Consideriamo 2 risorse lessicali:**

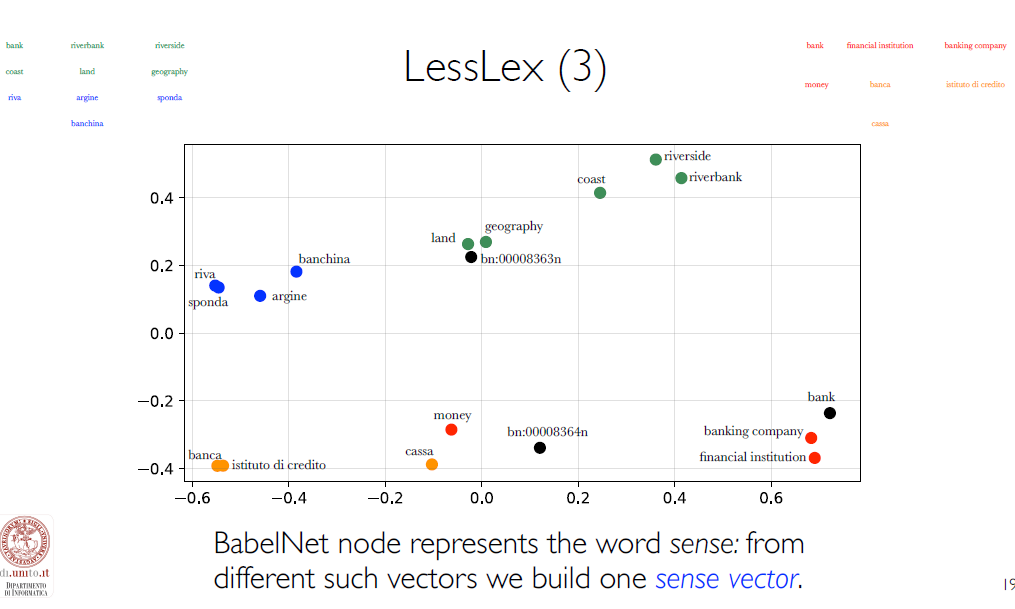
* **Babelnet**: E’ una rete semantica costruita automaticamente integrando la conoscenza tra wordnet e wikipedia e i synset di babelnet sono disponibili in 284 lingue.
* **ConceptNET Numberbatch**: Risorsa distribuzionale, associa termini a vettori numerici costruita automaticamente integrando la conoscenza di senso comune di conceptnet altre risorse distribuzionali. Questi vettori sono rappresentazioni di significati in 78 lingue che condividono lo stesso spazio, vuol dire che posso prendere il vettore inglese per “chair”, e per “sedia”, confrontarli fra loro ed avere un punteggio significativo. Questi vettori ammettono 300 dimensioni.
  + **Condividere lo stesso spazio** vuol dire: posso confrontarli fra loro (dimensione uguale) e ottengo un risultato significativo

**LESSLEX:**

A cosa ci serve una nuova risorsa? Le risorse che abbiamo visto presentano problemi. L’idea è costruire una risorsa che trae i vantaggi da entrambe le rappresentazioni della conoscenza. Obiettivo: costruire una risorsa basata su synset per trarre il vantaggio della specificità lessicale a cui aggiungiamo un vettore numerico perchè vogliamo mantenere la facilità di utilizzo delle risorse distribuzionali. Vogliamo inoltre costruire una rappresentazione del senso nello stesso spazio dei termini (stessa lunghezza e stesso senso)

Esempio: costruire la rappresentazione LESSLEX di “bank”:

1. Accediamo a babelnet e otteniamo 2 synset: “Istituto i credito”, “sponda del fiume” Le rappresentazioni dei sensi con babelnet restituiscono una serie di informazioni:
   1. **Synset** in una delle 284 lingue
   2. **Glossa**
   3. **Titolo** della pagina di wikipedia
2. Notiamo che ognuna di queste 3 sorgenti di informazione sono formate da termini di contenuto (parole nel synset, titolo di wikipedia, parole di contenuto nella glossa) Possiamo estrarre i termini di contenuto e rappresentare un senso come un insieme di termini. Consideriamo un synset in inglese e un synset in italiano. Un senso è un insieme di termini
3. Infine rappresentiamo ognuno di questi termini con un vettore distribuzionale con concept numberbatch. Prendo “bank”, vado su conceptnet e cerco il vettore.
4. Un synset è adesso un insieme di vettori di conceptnet numberbatch
5. Ognuno di questi vettori definisce un punto in uno spazio multidimensionale. Notiamo che i punti associati a “sponda del fiume” sono ben distanziati da quelli associati a “istituto di credito”.
6. Ogni senso ha una sua area nello spazio (cluster) e in un qualche modo possiamo dire che il centro dell’area, rappresentato dal centroide del cluster è il synset rappresentante sintetico del senso a cui ci vogliamo definire. Consideriamo per il calcolo del centroide anche il termine madre “bank”. Il centroide è calcolato come media di tutti i vettori in tutte le dimensioni.
7. Possiamo confrontare il vettore del centroide di “sponda del fiume “ e di “istituto di credito” con il vettore di bank e scoprire che istituto di credito è più vicino a bank.



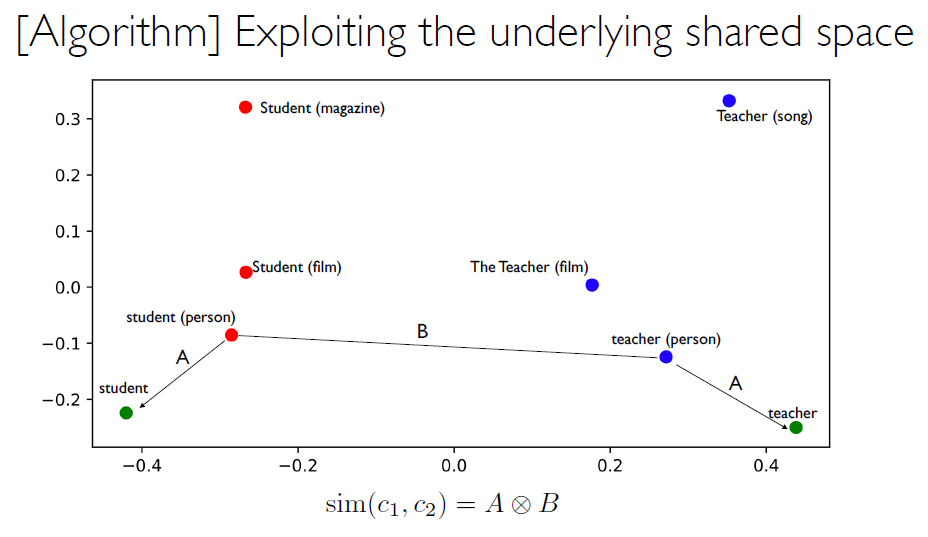
Perchè LESSLEX è stato costruito così?

* Rappresenta sensi a cui associo vettori distribuzionali
* La rappresentazione vettoriale avviene su 4 POS: Nomi, verbi, aggettivi, avverbi
* La risorsa mette assieme termini e sensi di lingue diverse nello stesso spazio semantico. Otteniamo un punteggio di similarità significativo
* Questa tecnica è un approccio generare tale da sostituirlo alle risorse lessicali e l’approccio continua a funzionare

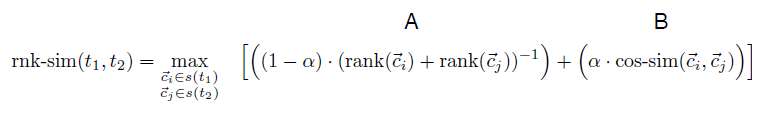
**Valutazione di LESSLEX:**

La risorsa funziona? Come si fa a capire? Si fa una valutazione. Bisogna prendere un problema appropriato per valutare la risorsa. Il task. Il task preso è **“Semantic similarity task”**: Dati 2 termini chiedo ai sistemi di formire un punteggio numerico che indichi quanto i 2 termini sono simili fra loro. Un punteggio tra 0 e 1. La risorsa di partenza è un dataset di coppie di termini (car, bicycle...) annotate con un valore di similarità. L’annotazione è fatta a mano da linguisti esperti. (**wordsyn353**). Usiamo LESSLEX per accedere alle rappresentazioni vettoriali dei sensi delle coppie di termini da confrontare. Ogni termine ha più babelsynset. Per ognuno di essi ricaviamo un vettore LESSLEX. Dalla lista di vettori calcoliamo la similarità del coseno a coppie. Restituiamo come similarità dei termini il massimo valore di similarità a coppie per tutte le coppie di vettori LESSLEX.

I risultati non sono soddisfacenti. Questo è dato dal fatto che ci sono coppie di sensi che spingono in su la similarità e che non sono i sensi a cui gli annotatori di wordsin353 di riferivano (Student(film), The Teacher(film)).



E’ stata riconsiderata la metrica di similarità usata per confrontare i vettori. Dapprima si sono individuati i punti LESSLEX più prossimi al vettore conceptnet Numberbatch del termine nella coppia, poi si è calcolata la similarità del coseno. La metrica che in una volta sola si permette di calcolare questa similarità è chiamata “max “ranked similarity”” e considera il rango dei synset. La distanza di un senso rispetto al termine è codificata dal rango, quanto è simile un senso rispetto al vettore terminologico.



La massimizzazione della ranked similarity funziona molto meglio rispetto alla massimizzazione della similarità del coseno. Infatti la correlazione delle similarità con la correlazione di pearson è molto forte.

**Sense Identification Task**

Qual è la procedura usata per formalizzare un nuovo task nella ricerca e come è stato risolto. Per risolvere un task abbiamo bisogno di una risorsa adeguata.

Ripartiamo dal task di conceptual similarity. Obiettivo, i sistemi approssimino meglio le similarità data dagli esseri umani.

Ciò che non coglie questa similarità è che quando asseriamo che esiste una certa somiglianza fra 2 termini ci focalizziamo sul senso più immediato. Quindi abbiamo la necessità di capire, data una coppia di termini per i quali vogliamo calcolare la similarità abbiamo bisogno di capire qual è il senso che ci viene in mente associato ai termini se presi entrambi. Quindi chiedere al sistema non di approssimare il punteggio di similarità dato dagli esseri umani ma di approssimare il più possibile i sensi che gli esseri umani ci hanno fornito per la coppia.

Per risolvere la similarità bisogna prima risolvere quali sensi vengono in mente quando si annotano 2 termini con la loro similarità.

Si parte dallo stesso dataset (wordsin) a cui assieme al valore di similarità aggiungiamo i sensi (synset) associati ai termini. Il processo di annotazione del dataset con i sensi dei termini è stato condotto chiedendo a diversi partecipanti di scrivere il senso che veniva loro in mente guardando la coppia di termini. L’annotazione ha visto una selezione dei partecipanti in base alle loro conoscenze di inglese.

In un primo approccio si nota che nel task di conceptual similarity applicando la massimizzazione della similarità a coppie di synset alla fine vince la coppia con lo score più alto, dunque si ricava anche la coppia di sensi e si risponde al task di sense identification. Sono state considerate 2 metriche:

* Massimizzazione della **similarità del coseno**
* Massimizzazione della **ranked similarity.**

La prima ammette lo stesso problema visto in precedenza. Si vanno a scegliere i sensi più vicini che tendenzialmente sono altri sensi che vanno fuori l’annotazione. Si può applicare la ranked similarity per confrontare il vettore ConceptNET del termine con il vettore LESSLEX dei sensi e individuare il senso più prossimo al termine inteso in un’accezione di vicinanza “termine-senso”, per ogni coppia di termini. Poi si calcola la similarità del coseno. La massimizzazione della ranked similarity ci fornisce un metodo di identificazione di sensi.

C’è un problema. La ranked similarity identifica coppie di sensi. Ci sono casi in cui è facile vedere una naturale aggregazione di sensi riferiti ad un certo termine in più cluster. Però notiamo che se prendiamo il vettore LESSLEX di weather inteso come condizione atmosferica e lo confrontiamo con tutti i vettori LESSLEX di wave usando la ranked similarity, abbiamo dei risultati che ordinati per valore decrescente ci rimandano ad una sorta di aggregazione per cluster dei sensi di wave.

Da questa intuizione è stata costruita una nuova metrica: “Neighbourhood similarity” la quale può essere utile sia per il task di conceptual similarity sia per il task di sense identification. A differenza degli approcci di clustering, la metrica fissa un senso e costruisce un vicinato di sensi.

**Neighbourhood similarity: funzionamento**

* Si parte dalla ranked similarity e si trovano i sensi più vicini al termine di riferimento, chiamati pivot
* Guardiamo nell’intorno di questi quali sono i possibili sensi vicini. Usiamo un parametro di confronto per capire chi sono i vicini più vicini.
* Definiamo la nozione di vicinato semantico come aggregazione dei sensi più vicini ai pivot

onda del mare può essere confuso con tutti i suoi vicini quando consideriamo la coppia weather e wave. E’ come se ammettessimo che nel processo di annotazione i partecipanti hanno detto che la nella coppia “weather – wave”, “wave” può essere inteso come tutti i sensi che stanno nel suo vicinato semantico.

* Calcoliamo il centroide per il vicinato semantico del pivot
* Calcoliamo la similarità del coseno tra i centroidi

**Valutazione: 3 approcci:**

* Max similarità del coseno
* Max ranked similarity
* Similarità del vicinato

Confrontiamo i punteggi dati dagli esseri umani con quelli restituiti dai 3 approcci, quindi calcoliamo la correlazione per i valori numerici, Precision e recall per i synset annotati. Si nota che la neigborhood si comporta meglio per la sense identification rispetto alla ranked similarity che è più adatta alla conceptual similarity.