Project 1

Jesus Soto Gonzalez
jhsoto@iastate.edu

Abstract

This first project for COM S 5790, asks us to apply basic NLP methods to a given dataset. We were required to preprocessed the text by tokenizing it, removing stopwords, and using phrase mining. From this, we created frequency- and TF–IDF-based word clouds to highlight key terms. Similarly, we were asked to train Word2Vec models on both words and phrases to find the nearest neighbors of top TF–IDF terms. Phrase mining with `gensim`'s bigram and trigram models helped capture multiword expressions and produced clearer, more meaningful embeddings.

1 Methods

1.1 Preprocessing

The dataset was filtered to only include abstracts with Category = 1. Each abstract was sentence-split, tokenized, lowercased, and cleaned by removing stopwords and non-alphabetic tokens using NLTK. The result was saved both as plain text for word clouds and as token lists in JSON for Word2Vec tasks respectively. This step ensured consistency across all the other tasks and reduced noise from irrelevant or low value tokens.

1.2 Phrase Mining

To capture multiword terms, I used the `gensim` library for phrase mining. I trained a bigram model on the tokenized corpus with a minimum count of 8 and a threshold of 8.0. This combined pairs of words that often appeared together. Then, I trained a trigram model on top of the bigrams to join longer expressions. In both cases, phrases were written with underscores, such as *carcass_composition*, *average_daily_gain*, and *backfat_thickness*.

I also tested different parameter values. Stricter settings 10, 10.0 created fewer but very strong phrases, while looser ones 5, 5.0 or 3, 3.0 created more phrases but added noise. The balanced setting 8, 8.0 worked the best, since it kept important domain phrases like *quantitative_trait_locus* and *meat_quality* without introducing too many random ones. This choice was important because phrase quality directly influenced the TF–IDF terms and the results from Word2Vec.

1.3 TF–IDF and Word Clouds

From both the single word and phrased corpus I computed frequency counts and TF–IDF scores. The top-ranked terms were visualized as word clouds, which made it easy to compare terms that are common against those that are more distinctive. This provided a clear way to see how phrase mining and weighting methods highlight different aspects of the corpus.

1.4 Word2Vec

I trained Word2Vec models on both the words and phrased corpus using CBOW with a vector size of 100, a window of 5, and a minimum count of 10 as required. Then, to evaluate the models, I looked at the nearest neighbors of the top TF–IDF terms and compared how the results changed when using phrases instead of single words.

2 Main Results

2.1 Word Clouds – Task 1

The frequency based word cloud had mostly common research terms such as *qtl*, *traits*, *study*, and *genes*. These words appear often across many abstracts but do not point to specific findings. In contrast, the TF–IDF word cloud highlighted more distinctive domain terms, including *ascites*, *earlobe*, *pleurisy*, and *ketosis*. This shows how TF–IDF can show more technical vocabulary that is less frequent overall but more informative for understanding the dataset.

(See Appendix, Figure 1).

2.2 Word2Vec – Task 2

As mentioned before, the Word2Vec model was trained on the corpus using CBOW with a vector size of 100, a window of 5, and a minimum count of 10. To evaluate the model, I looked at the nearest neighbors of the top TF–IDF terms.

One clear example was *ascites*, which refers to the buildup of fluid in the abdomen and is often used in a disease or health context (Cleveland Clinic, 2025). Its nearest neighbors included *death*, *vaccine*, *pathogens*, and *treatment*. These are all directly related to illness and medical outcomes, showing that the model was able to group words with a strong connection.

At the same time, some abbreviations such as *lp*, *ifc*, and *su* returned vague neighbors like *become* or *finally*, which were harder to interpret and less useful.

Overall, the results show that Word2Vec found useful connections for biological or health-related terms, but it struggled with abbreviations or other terms that did not occur often enough in the dataset.

2.3 Word Clouds – Task 3

Using the phrased corpus, the word clouds showed some multiword expressions that were not visible in Task 1. The frequency word cloud still contained mainly common research terms, but now phrases like *candidate_genes* appeared, giving more precise meaning than the single words alone.

The TF–IDF word cloud highlighted technical phrases, such as *subclinical_ketosis*. Terms like these capture specific meaning that would not have been clear if only individual words were used.

This improvement was a direct result of the phrase mining experiment in `phrases.py`, where I tested different parameter settings for `min_count` and `threshold`. Looser values produced many phrases but also added noise, and stricter values gave very few. The balanced choice (8, 8.0) worked best and helped the word clouds show some meaningful scientific terms instead of just single words.

(See Appendix, Figure 2).

2.4 Word2Vec – Task 3

The Word2Vec model was trained again on the phrased corpus. The same parameters were used: CBOW, vector size 100, window size 5, and minimum count 10. I evaluated it the same way as in Task 2, by looking at the nearest neighbors of the top TF–IDF terms.

Some results showed improvements. For example, *ascites* was now grouped with terms or technical phrases like *lesions* and *pregnancy_rate*, which are linked to health. This was more specific compared to the simpler words in Task 2.

Again, the balanced parameter setting of (8, 8.0) worked best compared to the others, since it directly influenced the phrased Word2Vec neighborhoods. It also showed real connections with technical meaning unlike Task 2.

However, even with phrases, not all terms were useful. Abbreviations like *lp* and *ifc* still produced neighbors that were hard to interpret, and some words like *abt* were skipped because of their low frequency.

Overall, using the phrased corpus made the neighbors more meaningful by showing technical multiword terms, especially in health contexts. On the other hand, rare abbreviations and terms were still limiting.

Finally, I compared all extracted phrases against the trait dictionary using exact string matching. Out of 22,719 trait entries, 282 matches were found. When the spaces were replaces with underscores to align with the phrased corpus, the number of matches increased to 348. This confirms that a significant amount of the phrases overlap with known traits, showing the method captured relevant domain terms.

Using the ACL template (Association for Computational Linguistics, 2025).

References

Association for Computational Linguistics. 2025. Acl style files. Accessed: September 28, 2025.

Cleveland Clinic. 2025. Ascites: What it is, causes, symptoms and treatment. Accessed: September 28, 2025.

Appendix

A (Word Cloud Figures)

Figure 1: Task 1 word clouds. Left: frequency-based, showing common terms such as *qtl* and *traits*. Right: TF–IDF-based, highlighting more distinctive terms like *ascites* and *earlobe*.

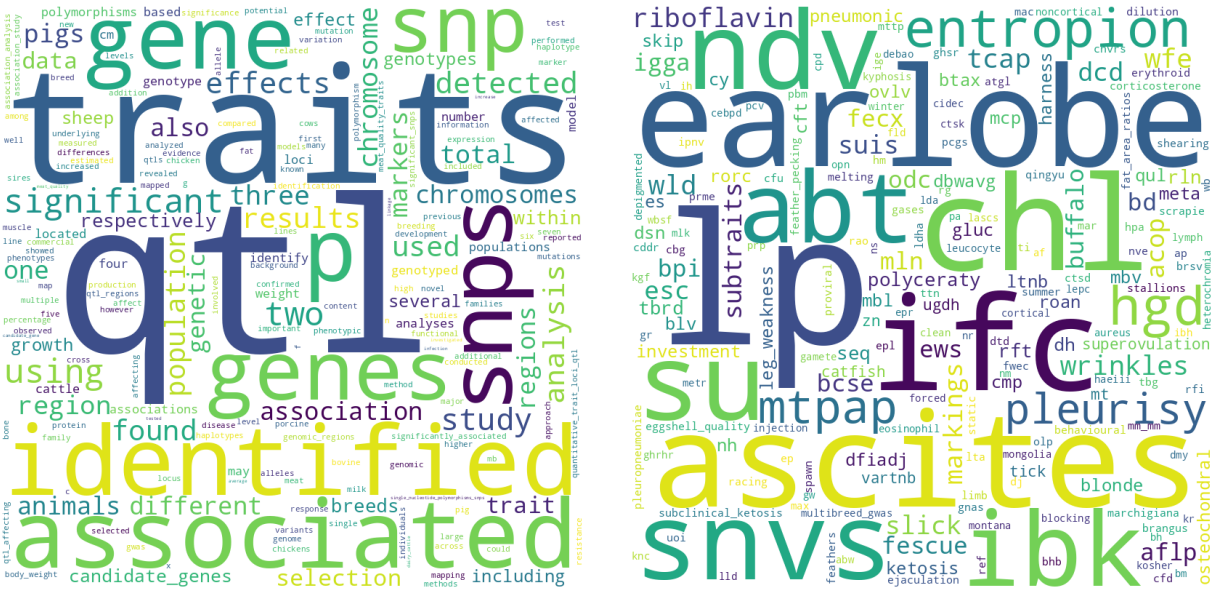

Figure 2: Task 3 word clouds with phrase mining. Left: frequency-based, where multiword terms such as *candidate_genes* appear. Right: TF–IDF-based, highlighting technical phrases like *subclinical_ketosis*.